VT20

DIAGNOSTIC TEST

MD-11-DBVTA-D

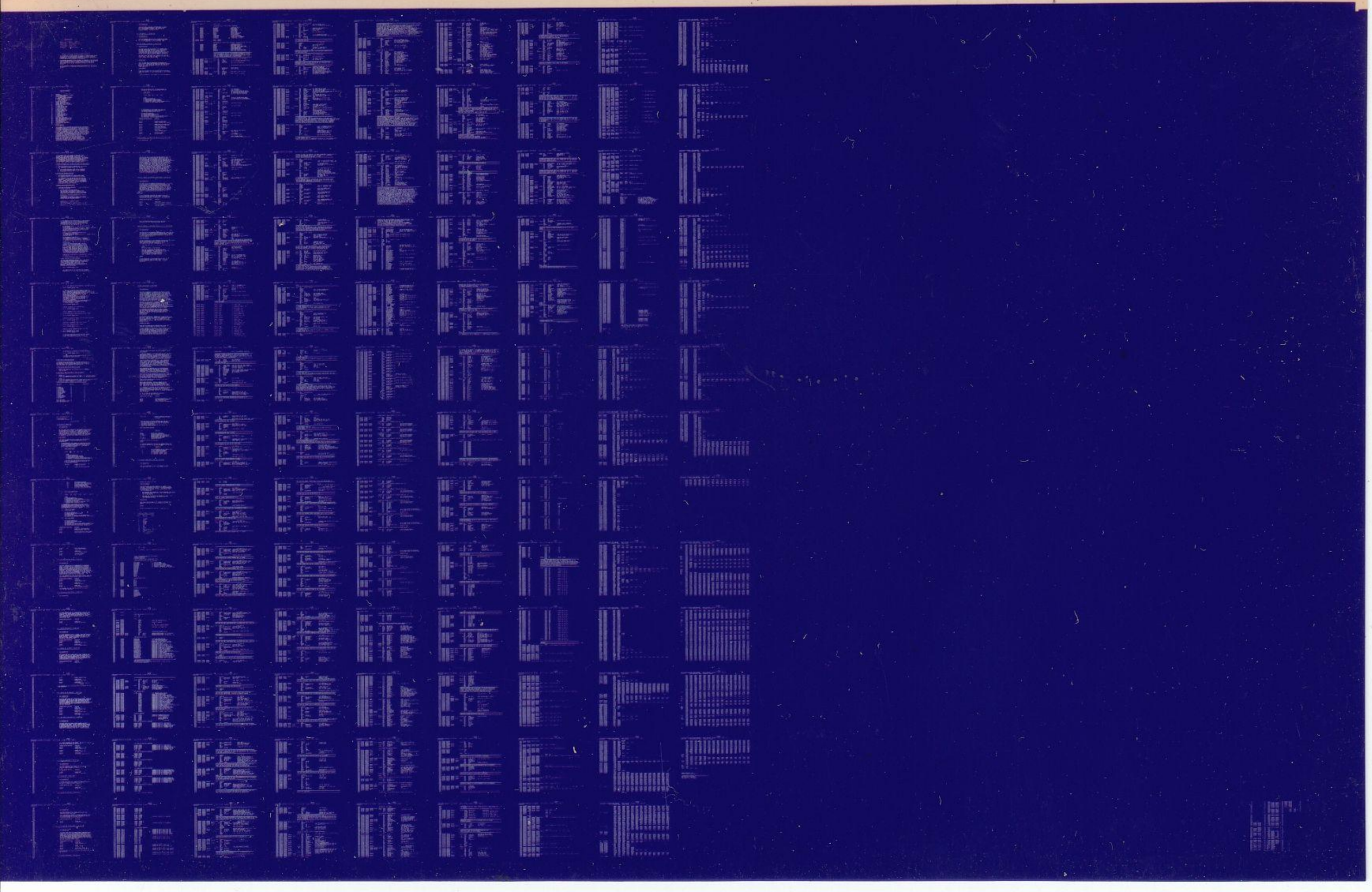
EP DBVTA-D-DL-B

FICHE 1 OF 1

MAR 1977

digital

MADE IN USA



.REM %

IDENTIFICATION

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

PRODUCT NAME: VT20 DIAGNOSTIC TEST

DATE RELEASED: JANUARY, 1977

MAINTAINER: DIAGNOSTIC GROUP

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. R E Q U I P E M E N T S (E Q U I P M E N T & M E M O R Y)

****

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. L O A D I N G P R O C E D U R E

****

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
   (175610 OR 175620). REFER TO THE PARTICULAR MAINDEC
   BEING USED FOR THE BOOT PROCEDURE.

4. S T A R T I N G P R O C E D U R E

****

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 AUTOMATICALL
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" - RESPONC

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 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 002062 (2064)

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. RESTART PROCEDURE

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. CONSOLE SWITCH SETTINGS

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

7. CONTROL SWITCHES (TELETYPE)

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.

8. PROGRAM TEST TABLE & DIRECTORY

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

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	RECV'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

SW12=0	NORMAL RUN
SW12=1	ADVANCE TO NEXT TEST

12. VT 'CSR PRESET' TEST (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

SW12=0	NORMAL RUN
SW12=1	ADVANCE TO NEXT TEST
SW14=0	NORMAL RUN
SW14=1	HALT ON CURRENT PRESET

13. VT 'END OF LINE' TEST (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

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=1	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 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 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

SW12=0	NORMAL RUN
SW12=1	ADVANCE TO NEXT TEST

19. VT 'CURSOR CONTROL' TEST (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

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 'ODD ADRESS' TEST (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 SW15 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

834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
954
855
856
857
858
859
860
861
862
863
864
865
866
867
968
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889

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 COMPLIMENTED 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

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

```
890
891                         NOTE: FOR PROPER OPERATION SW15 SHOULD BE RE-SET TO '0'
892                         AS SOON AS THE NEXT CHARACTER HAS BEEN INPUTED.
893
894
895
896             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)
897                     ************************************************************
898
899
900                     A.  TEST DESCRIPTION
901                         ----------------
902
903                         THIS TEST IS ALSO AN OPERATOR INTERVENTION TEST.  THE TEST HAS
904                         TWO FUNCTIONS.  IT CAN BE USED AS A BACKUP FOR THE 'CHARACTER
905                         SET' TEST TO TEST ANYONE SELECTED CHARACTER CODE OR, SECOND,
906                         IT CAN BE USED TO TEST KEYBOARD REPEATIBILITY BETWEEN ANY TWO
907                         SELECTED CHARACTER CODES.
908
909                     B.  STARTING SEQUENCE
910                         -----------------
911
912                         THE TEST VERIFIES UP TO TWO CHARACTERS RECEIVED FROM THE KEYBOARD
913                         AGAINST THE CHARACTER CODE(S) LOADED BY THE OPERATOR.
914                         THESE CHARACTER CODES ARE LOADED ONE OF TWO WAY, (1) VIA
915                         TELETYPE OR (2) VIA 'SWR' ON PROGRAM HALTS.
916
917                         1.  TYPE: NNN[,NNN] <CR> (I.E. 101,141  UPPER CASE A AND LOWER A)
918
919                             WHERE:
920                             NNN IS THE CODE(S) TO BE VERIFIED.
921
922                         2.  WHEN TEST IS SELECTED THE PROGRAM WILL HALT.
923                             LOAD THE CHARACTER CODE TO BE VERIFIED IN THE 'SWR'
924                             AND PRESS CONTINUE.  A SECOND HALT WILL OCCUR.
925                             IF TWO CHARACTERS ARE TO BE VERIFIED
926                             LOAD THE CODE OF THE SECOND CHARACTER IN THE
927                             'SWR', OTHERWISE JUST PRESS CONTINUE.
928
929                     C.  ERRORS
930                         ------
931
932                         IF THE TRANSMITTED CHARACTER CODE DOESN'T MATCH THE
933                         EXPECTED CHARACTER CODE THE AUDIO 'BEEP' IS SOUNDED
934                         AND THE RECEIVED CHARACTER IS DISPLAYED IN 'BOLD-
935                         UNDERLINED'.
936
937
```

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-08-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 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 CONTINOUS 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-DZVTG.

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 RESTART
↑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. E X E C U T I O N T I M E S

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.


                                    %



                                    .TITLE  VT20 SYSTEM DIAGNOSTIC TEST PROGRAM
                                    ;MAINDEC-11-DBVTA-D-D
                                    ;COPYRIGHT 1972,1976
                                    ;DIGITAL EQUIPMENT CORP. MAYNARD MASS. 01754

                                    ;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
005726                              POP1SP=5726
005746                              PUSH1SP=5746
024646                              PUSH2SP=24646
022626                              POP2SP=22626
000240                              NOP=240
000002                              X=2
```

```
1216            000002                  Y=2
1217
1218                                    ;LOAD TRAP CATCHER INTO LOC'S 0-1000
1219            000000                          .=0
1220                                    .REPT   200
1221                                            .+2
1222                                            4
1223                                    .ENDR
1224            000020                          .=20
1225  000020    016456                          ERTRAP                  ;ERROR TRAP REPORTER ROUTINE
1226            000024                          .=24
1227  000024    016014                          PWRFAL                  ;POWER FAIL HANDLER
1228  000026    000340                          340
1229            000030                          .=30
1230  000030    001200                          EMTSRV                  ;EMT TRAP, EMT DISPATCH SERVICE
1231  000032    000340                          340
1232  000034    017446                          FKERR                   ;TRAP TRAP, LOGIC ERROR TRAP
1233  000036    000340                          340
1234            000052                          .=52
1235  000052    020000                          20000
1236            000060                          .=60
1237  000060    014350                          XTTYIN                  ;TELEPRINTER KEYBOARD ROUTINE
1238            000200                          .=200
1239  000200    000137  001624                  JMP     INITTY          ;PROGRAM STARTING ADDR. IF TTY AVAILABLE.
1240  000204    000137  001634                  JMP     NOTTY           ;PROGRAM STARTING ADDR. IN 'NO' TTY.
1241  000210    000137  002136                  JMP     INITBA          ;PROGRAM RE-START ADDR. TO SELECT NEW UNIT.
1242
1243                                    ;TRAP EQUIVALENCE TABLE:
1244
1245          104400                    ERROR=TRAP                      ;LOGIC TEST ERROR ROUTINE
1246          104000                    DISPLAY=EMT                     ;'VT' MESSAGE DISPLAY ROUTINE
1247          104001                    SCOPE=EMT+1                     ;LOGIC TEST SCOPE SUBROUTINE
1248          104002                    SAVREG=EMT+2                    ;SUBROUTINE TO SAVE 'R0-R5' ON STACK
1249          104003                    GETREG=EMT+3                    ;SUBROUTINE TO GET 'R0-R5' FROM STACK
1250          104004                    DELAY=EMT+4                     ;3 SEC. DISPLAY DELAY
1251          104005                    EOSBUF=EMT+5                    ;SUBROUTINE TO LOAD VT BUFER W/ 'EOS'
1252          104006                    ENDTST=EMT+6                    ;SUBROUTINE TO CHECK DATA SW.'S
1253          104007                    SETLNE=EMT+7                    ;SUBROUTINE TO SET UP A LINE TO CHAR.'S
1254          104010                    LDLINE=EMT+10                   ;SUBROUTINE TO LOAD A 'VT' BUFFER LINE
1255          104011                    PRELOAD=EMT+11                  ;SUBROUTINE TO PRE-LOAD 'VT' DATA BUFFER
1256          104012                    PRINT=EMT+12                    ;SUBROUTINE TO PRINT ASCII MESSAGES.
1257          104013                    TTYIN=EMT+13                    ;SUBROUTINE TO INPUT VIA KEYBOARD
1258          104014                    PRTOCT=EMT+14                   ;SUBROUTINE TO PRINT A 6 DIGIT OCTAL NO.
1259          104015                    ASEMBL=EMT+15                   ;SUBROUTINE TO ASSEMLE CHARACTERS INTO OCTAL VALUE
1260          104016                    SPACE=EMT+16                    ;SUBROUTINE TO PRINT SPACES
1261          104017                    TSTTKS=EMT+17                   ;SUBROUTINE TO TEST FOR KEYBOARD FLAGS
1262          104020                    DELAYL=EMT+20                   ;SUBROUTINE TO SETUP A LONG DISPLAY DELAY
1263          104021                    WAITSY=EMT+21                   ;SUBROUTINE TO WAIT FOR VERTICAL SYNC
1264          104022                    NULL=EMT+22                     ;SUBROUTINE TO TRANSMIT A NULL PRINTER CHAR.
1265          104023                    THEND=EMT+23                    ;SUBROUTINE ENTERED AT END OF DISP. TEST
1266          104024                    BINDEC=EMT+24                   ;SUBROUTINE TO CONVERT 'BCD' TO DECIMAL
1267
1268                                    ;*****************************************************************************
1269                                    ;EMT DISPATCH SERVICE ROUTINE
1270                                    ;ARGUMENT OF EMT IS EXTRACTED AND USED AS OFFSET TO OBTAIN POINTER
1271                                    ;TO THE SELECTED SUBROUTINE.
```

```
1272                                    ;*************************************************************************
1273
1274            001200                          .=1200
1275  001200   011646          EMTSRV: MOV     (SP),-(SP)      ;GET PC FOR TO RETURN
1276  001202   162716  000002        SUB     #2,(SP)         ;PC OF EMT
1277  001206   017616  000000        MOV     @(SP),(SP)      ;GET EMT
1278  001212   005715                TST     (SP)            ;IS EMT VALID?
1279  001214   001001                BNE     EMTOK
1280  001216   000000                HALT                    ;INVALID EMT
1281  001220   006316          EMTOK:  ASL     (SP)            ;MULTIPLY EMT ARG BY '2'
1282  001222   042716  177001        BIC     #177001,(SP)    ;CLEAR UNWANTED BITS
1283  001226   062716  001240        ADD     #EMTTAB,(SP)    ;POINTER TO SUBROUTINE ADDRESS
1284  001232   017616  000000        MOV     @(SP),(SP)      ;SUBROUTINE ADDRESS
1285  001236   000136                JMP     @(SP)+          ;GO TO SUBROUTINE
1286
1287                                    ;EMT DISPATCH TABLE
1288
1289  001240   017246          EMTTAB: VTMES                   ;'VT' MESSAGE DISPLAY ROUTINE
1290  001242   020272                  SCOPEC                  ;LOGIC TEST SCOPE ROUTINE
1291  001244   016122                  XSAVRG                  ;SUBROUTINE TO SAVE 'R0-R5' ON STACK
1292  001246   016176                  XGETRG                  ;SUBROUTINE TO GET 'R0-R5' FROM STACK
1293  001250   015540                  XDELAY                  ;SUBROUTINE TO SETUP 3 SEC. DISPLAY DELAY
1294  001252   015656                  XEOSBF                  ;SUBROUTINE TO LOAD VT BUFFER W/ VISIBLE 'EOS'
1295  001254   015732                  XENDT                   ;SUBROUTINE TO CHECK DATA SW.'S
1296  001256   017024                  XSTLNE                  ;SUBROUTINE TO SET UP A 'VT' BUFFER LINE
1297  001260   015632                  XLDLNE                  ;SUBROUTINE TO LOAD A 'VT' BUFFER LINE
1298  001262   016252                  XPRELD                  ;SUBROUTINE TO PRE-LOAD THE DATA BUFFER
1299  001264   014754                  TYPMES                  ;SUBROUTINE TO PRINT ASCII MESSAGES.
1300  001266   014350                  XTTYIN                  ;SUBROUTINE TO INPUT VIA KEYBOARD
1301  001270   015056                  OCTPRT                  ;SUBROUTINE TO PRINT A 6 DIGIT OCTAL NO.
1302  001272   017104                  XASEMB                  ;SUBROUTINE TO ASSEMLE CHARACTERS INTO OCTAL VALUE
1303  001274   014276                  XSPACE                  ;SUBROUTINE TO PRINT SPACES
1304  001276   014330                  TKSFLG                  ;SUBROUTINE TO TEST FOR KEYBOARD FLAGS
1305  001300   015550                  XDLAYL                  ;SUBROUTINE TO SET UP LONG DISPLAY DELAY
1306  001302   015476                  XWAITS                  ;SUBROUTINE TO WAIT FOR VERT. SYNC
1307  001304   017360                  XNULL                   ;SUBROUTINE TO TRANSMIT A 'NULL' CHAR.
1308  001306   015726                  XTHEND
1309  001310   016546                  XBINDEC                 ;SUBROUTINE TO CONVERT 'BCD' TO DECIMAL
1310
1311                                    ;REGISTER ADDRESSES
1312
1313  001312   177776          PSW:    177776                  ;ADDRESS OF PROCESSOR STATUS REG.
1314  001314   177560          TKS:    177560                  ;ADDRESS OF KEYBOARD STATUS REG.
1315  001316   177562          TKB:    177562                  ;   "       "       "    BUFFER  "
1316  001320   177564          TPS:    177564                  ;   "       "   PRINTER STATUS REG.
1317  001322   177566          TPB:    177566                  ;   "       "   PRINTER BUFFER REG.
1318  001324   177570          SWR:    177570                  ;   "       "   SWITCH REG.
1319  001326   177571          SWR0:   177571                  ;   "       "       "       " HIGH BYTE
1320                                  :'FK' UNIT 0'S REGISTER ADDRESSES.
1321
1322  001330   170330          FKOLDB: 170330                  ;ADDRESS OF UNIT 0'S 'LIGHT DATA B'
1323  001332   170332          FKOLDA: 170332                  ;ADDRESS OF UNIT 0'S 'LIGHT DATA A'
1324  001334   170334          FKOCSR: 170334                  ;ADDRESS OF UNIT 0'S 'CONTROL & STATUS' REG.
1325  001336   170336          FKODAT: 170336                  ;ADDRESS OF UNIT 0'S 'KEYBORAD DATA' REG.
1326  001340   000320          FKOINT: 320                     ;ADDRESS OF UNIT 0'S 'INTERRUPT VECTOR'
1327  001342   000322          FKOLVL: 322                     ;ADDRESS OF UNIT 0'S 'INTERRUPT LEVEL'
```

```
1328
1329                                    ;'FK' UNIT 1'S REGISTER ADDRESSES.
1330
1331  001344  170340                    FK1LDB: 170340                  ;ADDRESS OF UNIT 1'S 'LIGHT DATA B'
1332  001346  170342                    FK1LDA: 170342                  ;ADDRESS OF UNIT 1'S 'LIGHT DATA A'
1333  001350  170344                    FK1CSR: 170344                  ;ADDRESS OF UNIT 1'S 'CONTROL & STATUS' REG.
1334  001352  170346                    FK1DAT: 170346                  ;ADDRESS OF UNIT 1'S 'KEYBOARD DATA' REG.
1335  001354  000324                    FK1INT: 324                     ;ADDRESS OF UNIT 1'S 'INTERRUPT VECTOR'
1336  001356  000326                    FK1LVL: 326                     ;ADDRESS OF UNIT 1'S 'INTERRUPT LEVEL'
1337
1338                                    ;'FK' UNIT 2'S REGISTER ADDRESSES
1339
1340  001360  170360                    FK2LDB: 170360
1341  001362  170362                    FK2LDA: 170362
1342  001364  170364                    FK2CSR: 170364
1343  001366  170366                    FK2DAT: 170366
1344  001370  000350                    FK2INT: 350
1345  001372  000352                    FK2LVL: 352
1346
1347                                    ;'FK' UNIT 3'S REGISTER ADDRESSES
1348
1349  001374  170370                    FK3LDB: 170370
1350  001376  170372                    FK3LDA: 170372
1351  001400  170374                    FK3CSR: 170374
1352  001402  170376                    FK3DAT: 170376
1353  001404  000354                    FK3INT: 354
1354  001406  000356                    FK3LVL: 356
1355
1356                                    ;'VT' UNIT 0'S REGISTER ADDRESSES.
1357
1358  001410  170300                    VT0CAR: 170300                  ;ADDRESS OF UNIT 0'S CURSOR ADDRESS REG.
1359  001412  170302                    VT0SAR: 170302                  ;ADDRESS OF UNIT 0'S STATING ADDRESS REG.
1360  001414  170304                    VT0CSR: 170304                  ;ADDRESS OF UNIT 0'S COMMAND & STATUS REG.
1361  001416  170306                    VT0MAR: 170306                  ;ADDRESS OF UNIT 0'S MAINTENANCE REG.
1362  001420  000330                    VT0INT: 330                     ;ADDRESS OF UNIT 0'S INTERRUPT VECTOR
1363  001422  000332                    VT0LVL: 332                     ;ADDRESS OF UNIT 0'S INTERRUPT LEVEL
1364
1365                                    ;'VT' UNIT 1'S REGISTER ADDRESSES.
1366
1367  001424  170310                    VT1CAR: 170310                  ;ADDRESS OF UNIT 1'S CURSOR ADDRESS REG.
1368  001426  170312                    VT1SAR: 170312                  ;ADDRESS OF UNIT 1'S STATING ADDRESS REG.
1369  001430  170314                    VT1CSR: 170314                  ;ADDRESS OF UNIT 1'S COMMAND & STATUS REG.
1370  001432  170316                    VT1MAR: 170316                  ;ADDRESS OF UNIT 1'S MAINTENANCE REG.
1371  001434  000340                    VT1INT: 340                     ;ADDRESS OF UNIT 1'S INTERRJPT VECTOR
1372  001436  000342                    VT1LVL: 342                     ;ADDRESS OF JNIT 1'S INTERRUPT LEVEL
1373
1374                                    ;'VT' UNIT 2'S REGISTER ADDRESSES
1375
1376  001440  170320                    VT2CAR: 170320
1377  001442  170322                    VT2SAR: 170322
1378  001444  170324                    VT2CSR: 170324
1379  001446  170326                    VT2MAR: 170326
1380  001450  000360                    VT2INT: 360
1381  001452  000362                    VT2LVL: 362
1382
1383                                    ;'VT' UNIT 3'S REGISTER ADDRESSES
```

```
1384
1385  001454  170350          VT3CAR: 170350
1386  001456  170352          VT3SAR: 170352
1387  001460  170354          VT3CSR: 170354
1388  001462  170356          VT3MAR: 170356
1389  001464  000370          VT3INT: 370
1390  001466  000372          VT3LVL: 372
1391
1392  001470  170330          FKLDB:  170330
1393  001472  170332          FKLDA:  170332
1394  001474  170334          FKCSR:  170334          ;TEMPORARY STORAGE OF FK ADDRESSES
1395  001476  170336          FKDATA: 170336
1396  001500  000320          FKINT:  320
1397  001502  000322          FKLVL:  322
1398
1399  001504  170300          VTCAR:  170300
1400  001506  170302          VTSAR:  170302
1401  001510  170304          VTCSR:  170304          ;TEMPORARY STORAGE OF VT ADDRESSES
1402  001512  170306          VTMAR:  170306
1403  001514  000330          VTINT:  330
1404  001516  000332          VTLVL:  332
1405
1406                          ;DL11 REGISTER ADDRESSFS
1407
1408  001520  175610          RCSR0:  175610          ;ADDRESS OF UNIT 0'S DL11 REC. CSR
1409  001522  175620          RCSR1:  175620          ;ADDRESS OF UNIT 1'S DL11 REC. CSR
1410  001524  175630          RCSR2:  175630          ;ADDRESS OF UNIT 2'S DL11 REC. CSR
1411  001526  175640          RCSR3:  175640          ;ADDRESS OF UNIT 3'S DL11 REC. CSR
1412  001530  175612          RBUF0:  175612          ;ADDRESS OF UNIT 0'S DL11 REC. BUFFER
1413  001532  175622          RBUF1:  175622          ;ADDRESS OF UNIT 1'S DL11 REC. BUFFER
1414  001534  175632          RBUF2:  175632
1415  001536  175642          RBUF3:  175642
1416  001540  000300          RINT0:  300             ;ADDRESS OF UNIT 0'S REC. VECTOR
1417  001542  000302          RLVL0:  302
1418  001544  000310          RINT1:  310             ;ADDRESS OF UNIT 1'S REC. VECTOR
1419  001546  000312          RLVL1:  312
1420  001550  000400          RINT2:  400
1421  001552  000402          RLVL2:  402
1422  001554  000410          RINT3:  410
1423  001556  000412          RLVL3:  412
1424
1425  001560  175614          XCSR0:  175614          ;ADDRESS OF UNIT 0'S TRANS. CSR
1426  001562  175624          XCSR1:  175624          ;ADDRESS OF UNIT 1'S DL11 TRANS. CSR
1427  001564  175634          XCSR2:  175634
1428  001566  175644          XCSR3:  175644
1429  001570  175616          XBUF0:  175616          ;ADDRESS OF UNIT 0'S DL11 TRANS. BUFFER
1430  001572  175626          XBUF1:  175626          ;ADDRESS OF UNIT 1'S DL11 TRANS. BUFFER
1431  001574  175636          XBUF2:  175636
1432  001576  175646          XBUF3:  175646
1433  001600  000304          XINT0:  304             ;ADDRESS OF UNIT 0'S DL11 TRANS. VECTOR
1434  001602  000306          XLVL0:  306
1435  001604  000314          XINT1:  314             ;ADDRESS OF UNIT 1'S DL11 TRANS. VECTOR
1436  001606  000316          XLVL1:  316
1437  001610  000404          XINT2:  404
1438  001612  000406          XLVL2:  406
1439  001614  000414          XINT3:  414
```

```
  1440  001616  000416                XLVL3:  416
  1441
  1442                                ;DEFINITIONS OF 'VT FIELD CONTROL' FUNCTIONS
  1443
  1444          000370                UNLINE=370                    ;UNDERLINE PRESET
  1445          000364                BOLD=364                      ;BOLD PRESET
  1446          000361                BLANK=361                     ;BLANK PRESET
  1447          000362                BLINK=362                     ;BLINK PRESET
  1448          000200                AUDIO=200                     ;AUDIO PRESET
  1449          000012                EOL=12                        ;END OF LINE
  1450          000014                EOP=14                        ;END OF PARAGRAPH
  1451          000031                EOS=31                        ;END OF SCREEN
  1452          000360                CLRFLD=360                    ;CLR FIELD CONTROL
  1453          000212                VISEOL=212                    ;VISIBLE END OF LINE
  1454          000214                VISEOP=214                    ;VISIBLE END OF PARAGRAPH
  1455          000231                VISEOS=231                    ;VISIBLE END OF SCREEN
  1456
  1457  001620  000000                FCSET:  000000
  1458  001622  000000                FCSET1: 000000
  1459
  1460                                ;DEFINITIONS OF 'MAINTENANCE' FUNCTIONS ( SELECTED VIA THE 'CSR')
  1461
  1462          000000                CSR=000                       ;COMMAND & STATUS REG.
  1463          000002                DAR=2                         ;DATA ADDRESS REG.
  1464          000004                SRIDE=4                       ;SHIFT REG. INPUT DATA (EVEN)
  1465          000006                SRODE=6                       ;SHIFT REG. OUTPUT DATA (EVEN)
  1466          000010                CAR=10                        ;CURSOR ADDRESS REG.
  1467          000012                SAR=12                        ;START ADDRESS REG.
  1468          000014                SRIDO=14                      ;SHIFT REG. INPUT DATA (ODD)
  1469          000016                SRODO=16                      ;SHIFT REG. OUTPUT DATA (ODD)
  1470
  1471                                ;*********************************************************************
  1472                                ;TEST INITIALIZATION ROUTINE. IF THE PROGRAM IS STARTED AT LOCATION
  1473                                ;'200' THE PROGRAM IS SET UP TO RUN VIA A TELETYPE MONITOR. IF THE PRO-
  1474                                ;GRAM IS  STARTED AT LOCATION '204' THE PROGRAM USES PROGRAM 'HALTS'
  1475                                ;TO COLLECT NEEDED INFORMATION AND TO REPORT ERRORS.
  1476                                ;*********************************************************************
  1477
  1478  001624  012737  000001  024154  INITTY: MOV     #1,TTYSWH     ;TTY AVAILABLE, SET SOFTWARE SW.
  1479  001632  000402                  BR      .+6
  1480  001634  005037  024154          NOTTY:  CLR     TTYSWH        ;CLR TTY AVAILABLE SOFTWARE SWITCH.
  1481  001640  012777  000340  177444          MOV     #340,@PSW
  1482  001646  013706  024132                  MOV     STACK,SP      ;INIT STACK POINTER=1000
  1483  001652  000240                          NOP
  1484  001654  000240                          NOP
  1485  001656  000240                          NOP                   ;INITIALIZATION SWITCH TEST REMOVED TO ALLOW RE-SELECT
  1486  001660  000240                          NOP
  1487  001662  000240                          NOP
  1488  001664  012737  001702  000004  1$:     MOV     #CORSIZ,@#4   ;RESET TIMEOUT
  1489  001672  012701  020000                  MOV     #20000,R1     ;TEST CORE SIZE
  1490  001676  005721                          TST     (R1)+
  1491  001700  000776                          BR      .-2
  1492  001702  162701  002000          CORSIZ: SUB     #2000,R1      ;SAVE LAST 1K FOR 'VT' TEST
  1493  001706  010137  024140                  MOV     R1,MEMSIZ
  1494  001712  012737  000006  000004          MOV     #6,@#4        ;RESET TIMEOUT TO 'ERTRAP'
  1495  001720  012737  000004  000006          MOV     #4,@#6
```

```
1496  001726  005737  024154          INITB:  TST     TTYSWH          ;TTY AVAILABLE?
1497  001732  001106                          BNE     INITC           ;YES, INIT VIA TTY.
1498  001734  000000                          HALT                    ;SEE IF FORIEGN CHAR SET EXISTS
1499  001736  000240                          NOP                     ;BY EXAMINING THE SWITCH REGISTER
1500  001740  017737  177360  001620          MOV     @SWR,FCSET      ;IF CLEAR THEN NOT FORIEGN SET
1501  001746  000000                          HALT                    ;GET # OF TUBES FROM SWITCHES
1502  001750  000240                          NOP
1503  001752  017737  177346  024136          MOV     @SWR,   UNITNO
1504  001760  042737  177770  024136          BIC     #177770,UNITNO
1505  001766  005777  177332                  TST     @SWR            ;IF BIT 15 IN SWR IS SET THEN
1506  001772  100402                          BMI     4$              ;LET USER INPUT DEVICE ADDRESSES
1507  001774  000137  002436                  JMP     INITCB          ;DEFAULT DEVICES & VECTORS REQJESTED
1508
1509  002000  013700  024136          4$:     MOV     UNITNO,R0
1510  002004  012705  001330                  MOV     #FKOLDB,R5
1511  002010  000000                  1$:     HALT                    ;HALT FOR FK VECTOR
1512  002012  017737  177306  001500          MOV     @SWR,FKINT
1513  002020  000240                          NOP
1514  002022  000000                          HALT                    ;HALT FOR FK ADDR.
1515  002024  017704  177274                  MOV     @SWR,R4
1516  002030  004737  002700                  JSR     PC,SETED
1517  002034  005300                          DEC     R0
1518  002036  001364                          BNE     1$
1519
1520  002040  012705  001410                  MOV     #VTOCAR,R5
1521  002044  013700  024136                  MOV     UNITNO,R0
1522  002050  000000                  2$:     HALT                    ;HALT FOR VT VECTOR
1523  002052  017737  177246  001500          MOV     @SWR,FKINT
1524  002060  000240                          NOP
1525  002062  000000                          HALT                    ;HALT FOR VT ADDR.
1526  002064  017704  177234                  MOV     @SWR,R4
1527  002070  004737  002700                  JSR     PC,SETED
1528  002074  005300                          DEC     R0
1529  002076  001364                          BNE     2$
1530  002100  013700  024136                  MOV     UNITNO,R0
1531  002104  012702  001540                  MOV     #RINTO,R2
1532  002110  012705  001600                  MOV     #XINTO,R5
1533  002114  000000                  3$:     HALT                    ;HALT FOR DL11 VECTOR
1534  002116  017703  177202                  MOV     @SWR,R3
1535  002122  012322                          MOV     (3)+,(2)+
1536  002124  012322                          MOV     (3)+,(2)+
1537  002126  012325                          MOV     (3)+,(5)+
1538  002130  012325                          MOV     (3)+,(5)+
1539  002132  005300                          DEC     R0
1540  002134  001367                          BNE     3$
1541  002136  000000          INITBA: HALT                    ;GET TUBE NO. FROM SWITCHES
1542  002140  017737  177160  024134          MOV     @SWR,UNITFG     ;SAVE TUBE NO.
1543  002146  000543                          BR      INITD
1544  002150  005737  024166          INITC:  TST     MTRSWH          ;TYPE HEADER
1545  002154  001002                          BNE     .+6             ;NO, ITS BEEN TYPED
1546  002156  104012                          PRINT
1547  002160  021027                          TITLE                   ;PRINT PROGRAM HEADER
1548  002162  104012                          PRINT                   ;SEE IF FORIEGN CHARACTER SET EXISTS
1549  002164  021440                          MFCP
1550  002166  104015                          ASEMBL
1551  002170  042703  177776                  BIC     #177776,R3
```

```
1552  002174  010337  001620                  MOV     R3,FCSET
1553  002200  104012                          PRINT                   ;REQUEST NO. OF TUBES
1554  002202  024002                          MED1
1555  002204  104015                          ASEMBL                  ;GET INPUT
1556  002206  010337  024136                  MOV     R3,UNITNO
1557  002212  104012  021160                  PRINT,  MES1A           ;TEXT 'ARE DEFAULT ADDR. & VECT. OK?'
1558  002216  104015                          ASEMBL                  ;WAIT FOR HIS ANSWER
1559  002220  005703                          TST     R3              ;WAS 'CR' TYPED?
1560  002222  001505                          BEQ     INITCB          ;YES, USE DEFAULT ADDRESSES
1561  002224  022703  000001                  CMP     #1,R3           ;WAS 'Y' TYPED?
1562  002230  001502                          BEQ     INITCB          ;YES, IT SURE WAS
1563  002232  012705  001330          INITAC: MOV     #FKOLDB,R5      ;NO, LET USER DEFINE HIS OWN ADDRESSES
1564  002236  013700  024136          1$:     MOV     UNITNO,R0       ;WE HAVE ASK HIM FOR ADDRESSES
1565  002242  112737  000260  014750          MOVB    #260,MRUNN1
1566  002250  104012                  2$:     PRINT                   ;FIRST GET FK ADDRESS
1567  002252  024033                          MFKP
1568  002254  104012                          PRINT                   ;PRINT UNIT NO
1569  002256  014750                          MRUNN1
1570  002260  104015                          ASEMBL
1571  002262  010337  001500                  MOV     R3,FKINT
1572  002266  001770                          BEQ     2$
1573  002270  004737  002700                  JSR     PC,SETED
1574  002274  105237  014750                  INCB    MRUNN1
1575  002300  005300                          DEC     R0
1576  002302  001362                          BNE     2$
1577  002304  012705  001410                  MOV     #VTOCAR,R5      ;GET VT VECTORS AND ADDRESS
1578  002310  013700  024136                  MOV     UNITNO,R0       ;NUMBER OF TUBES
1579  002314  112737  000260  014750          MOVB    #260,MRUNN1
1580
1581  002322  104012                  3$:     PRINT
1582  002324  024026                          MUTP
1583  002326  104012                          PRINT
1584  002330  024037                          MAUFK
1585  002332  104012                          PRINT
1586  002334  014750                          MRUNN1
1587  002336  104015                          ASEMBL
1588  002340  010337  001500                  MOV     R3,FKINT
1589  002344  001766                          BEQ     3$
1590  002346  004737  002700                  JSR     PC,SETED
1591  002352  105237  014750                  INCB    MRUNN1
1592  002356  005300                          DEC     R0
1593  002360  001360                          BNE     3$
1594
1595  002362  013700  024136                  MOV     UNITNO,R0       ;ASk FOR DL11 VECTORS
1596  002366  012702  001540                  MOV     #RINT0,R2
1597  002372  012705  001600                  MOV     #XINT0,R5
1598  002376  112737  000260  014750          MOVB    #260,MRUNN1
1599
1600  002404  104012                  4$:     PRINT                   ;PRINT MESSAGE
1601  002406  024075                          MDLVA
1602  002410  104012                          PRINT
1603  002412  014750                          MRUNN1
1604  002414  104015                          ASEMBL
1605  002416  012322                          MOV     (3)+,(2)+
1606  002420  012322                          MOV     (3)+,(2)+
1607  002422  012325                          MOV     (3)+,(5)+
```

```
1608  002424  012325                          MOV     (3)+,(5)+
1609  002426  105237  014750                  INCB    MRUNN1
1610  002432  005300                          DEC     R0
1611  002434  001363                          BNE     4$
1612
1613  002436  005737  024154          INITCB: TST     TTYSWH
1614  002442  001635                          BEQ     INITBA
1615  002444  104012                          PRINT
1616  002446  021376                          MESS                    ;REQUEST UNIT NO.
1617  002450  104015                          ASEMBL                  ;DECODE IT.
1618  002452  010337  024134                  MOV     R3,UNITFG       ;SAVE IT
1619  002456  013737  024134  024126  INITD: MOV     UNITFG,PIMP
1620  002464  005037  024164                  CLR     SWAPEM          ;CLR SWAP SW.
1621  002470  022737  000004  024134          CMP     #4,UNITFG
1622  002476  001004                          BNE     .+12
1623  002500  005037  024134                  CLR     UNITFG          ;YES, SET UP UNIT '0'
1624  002504  005237  024164                  INC     SWAPEM          ;SET SWAP SW.
1625  002510  004737  016320                  JSR     PC,STUNIT       ;SET UP SELECTED UNIT ADDRESSES
1626  002514  005737  024154                  TST     TTYSWH          ;TTY AVAILABLE?
1627  002520  001402                          BEQ     MONITR          ;NO
1628  002522  104012                          PRINT                   ;YES,
1629  002524  021263                          MES3                    ;TEXT 'TYPE IN TST NO. TO BE EXECUTED'.
1630                                ;*************************************************************************
1631                                ;TEST SELECTION ROUTINE. IF NO TTY IS AVAILBLE THE PROGRAM WILL HALT
1632                                ;AND WAIT FOR A TEST NO. TO BE LOADED VIA THE SW.'S. IF A TTY IS
1633                                ;AVAIBLE, A REQUEST IS MADE VIA THE PRINTER FOR THE SELECTED TEST
1634                                ;NUMBER TO BE TYPED IN.
1635                                ;*************************************************************************
1636
1637  002526  000005                  MONITR: RESET                   ;INITIALIZE ON ENTRY
1638  002530  104022                          NULL                    ;NULL FOR PRINTER.
1639  002532  013706  024132                  MOV     STACK,SP        ;RESET STACK POINTER
1640  002536  005037  024164                  CLR     SWAPEM
1641  002542  022737  000004  024126          CMP     #4,PIMP         ;DOES HE WANT TO SWAP?
1642  002550  001006                          BNE     1$
1643  002552  005037  024134                  CLR     UNITFG
1644  002556  005237  024164                  INC     SWAPEM
1645  002562  004737  016320                  JSR     PC,STUNIT
1646  002566  012737  001726  024130  1$:     MOV     #INITB,AVECTR   ;SET UP '^A' RESTART ADDR.
1647  002574  004737  016710                  JSR     PC,CLRFKV       ;CLR 'FK' INTR ADDR TO HALT
1648  002600  004737  016774                  JSR     PC,CLRVTV       ;CLR 'VT' INTR ADDR TO HALT
1649  002604  012737  000001  024166          MOV     #1,MTRSWH       ;SET MONITOR SW.
1650  002612  005037  024150                  CLR     RUNSWH          ;CLR 'RUN ALL' SOFTWARE SW.
1651  002616  005037  024240                  CLR     HOLDSO
1652  002622  005037  024162                  CLR     DONESW
1653  002626  005737  024154                  TST     TTYSWH          ;TTY AVAILABLE?
1654  002632  001004                          BNE     TTYMTR          ;YES, REQUEST TEST NO.
1655  002634  000000                          HALT                    ;GET TEST NO. FROM DATA SW'S
1656  002636  017703  176462                  MOV     @SWR,R3         ;SAVE TEST NO.
1657  002642  000403                          BR      TTYMT1          ;EXECUTE SELECTED TEST
1658  002644  104012                  TTYMTR: PRINT
1659  002646  022774                          CNTRLC
1660  002650  104015                          ASEMBL                  ;DECODE IT
1661  002652  042703  177740          TTYMT1: BIC     #177740,R3      ;CLR UN-WANTED BITS
1662  002656  006303                          ASL     R3
1663  002660  020327  000044                  CMP     R3,#44
```

```
1664  002664  003002                          BGT     .+6             ;TYPE '?' IF NUMBER TO BIG
1665  002666  000173  002750                  JMP     @TESTOO(R3)     ;EXECUTE SELECTED TEST
1666  002672  104012                          PRINT
1667  002674  023007                          QMARK                   ;TYPE '?' ON ILLEGAL NO.'S
1668  002676  000714                          BR      MONITR+2        ;RE-START
1669  002700  010146                  SETED:  MOV     R1,-(6)         ;SAVE R1
1670  002702  012701  000004                  MOV     #4,R1
1671  002706  010425                  1$:     MOV     R4,(5)+         ;MOV ADDR TO ADDR. POINTER
1672  002710  062704  000002                  ADD     #2,R4
1673  002714  005301                          DEC     R1
1674  002716  001373                          BNE     1$
1675  002720  013725  001500                  MOV     FKINT,(5)+      ;MOV VECTOR ADDR TO ADDR POINTER
1676  002724  062737  000002  001500          ADD     #2,FKINT
1677  002732  013725  001500                  MOV     FKINT,(5)+
1678  002736  012601                          MOV     (6)+,R1
1679  002740  062737  000002  001500          ADD     #2,FKINT
1680  002746  000207                          RTS     PC
1681                                  ;*****************************************************************
1682                                  ;'VT20' TEST TABLE
1683                                  ;*****************************************************************
1684
1685  002750  003122                  TESTOO: VTLOGIC                 ;VT 'LOGIC' TEST
1686
1687  002752  005166                  TEST01: VTCHAR                  ;VT 'CHARACTER DISPLAY' TEST
1688
1689  002754  005226                  TEST02: VTMOD1                  ;VT 'FIELD MODE' TEST '1'
1690
1691  002756  005322                  TEST03: VTMOD2                  ;VT 'CSR PRESET' TEST
1692
1693  002760  005454                  TEST04: VTEOL                   ;VT 'END OF LINE' TEST
1694
1695  002762  005574                  TEST05: VTEOS                   ;VT 'END OF SCREEN' TEST
1696
1697  002764  005660                  TEST06: VTBLANK                 ;VT 'BLANK CONTROL' TEST
1698
1699  002766  005740                  TEST07: VTALINE                 ;VT 'ALIGNMENT' TEST
1700
1701  002770  005766                  TEST10: VTFOCUS                 ;VT 'FOCUS' TEST
1702
1703  002772  006014                  TEST11: VTWORST                 ;VT 'WORST CASE' TEST
1704
1705  002774  006042                  TEST12: VTCURSR                 ;VT 'CURSOR CONTROL' TEST
1706
1707  002776  006136                  TEST13: VTODD                   ;VT 'ODD ADDRESS' TEST.
1708
1709  003000  003016                  TEST14: VTRUN                   ;VT 'RUN ALL' TEST
1710
1711  003002  006236                  TEST15: FKLOGIC                 ;FK 'LOGIC' TEST
1712
1713  003004  007416                  TEST16: FKFUN                   ;FK 'FUNCTION' TEST
1714
1715  003006  007576                  TEST17: FKCHAR                  ;FK 'CHARACTER' TEST
1716
1717  003010  010126                  TEST20: FKREPT                  ;FK 'REPEATIBILITY' TEST
1718
1719  003012  010344                  TEST21: SYSTST                  ;VT 'SYSTEM EXERCISER' TEST
```

```
1720
1721  003014  014162                TEST22: TESTX                   ;FK/VT 'LOGIC SUBTEST' SELECTION
1722
1723                                ;*****************************************************************
1724                                ;SUBROUTINE  ENTERED VIA SELECTING 'VTRUN'.THIS ROUTINE SETS A
1725                                ;SOFTWARE SW. WHICH ENABLES THE VT LOGIC TEST, ALL DISPLAY TESTS, AND
1726                                ;THE ASCII KEYBOARD TEST TO BE RUN IN ORDER. THESE TESTS WILL BE RUN
1727                                ;ON BOTH UNITS IF UNIT '2' WAS SELECTED.
1728                                ;*****************************************************************
1729
1730  003016  005237  024150        VTRUN:  INC     RUNSWH          ;SET SOFTWARE SW.
1731  003022  000137  003122                JMP     VTLOGIC         ;START 1ST TEST.
1732                                ;*****************************************************************
1733                                ;VT20 LOGIC TEST (00)
1734                                ;*****************************************************************
1735
1736  003026  013704  001510        SETUPV: MOV     VTCSR,VTCSR4    ;LOAD 'R4' WITH THE 'CSR' ADDRESS
1737  003032  013705  001512                MOV     VTMAR,VTMAR5    ;LOAD 'R5' WITH THE 'MAINT' ADDR. REG.
1738  003036  005037  024444                CLR     MESPRT          ;CLR PRINT INHIBIT SW.
1739  003042  005037  024446                CLR     MESPT1          ;CLR PRINT INHIBIT SW.
1740  003046  012737  000100  020402        MOV     #100,ICOUNT
1741  003054  012737  017532  000034        MOV     #VTERR,@#34     ;LOAD THE TRAP 'TRAP' WITH LOGIC ERROR TRAP
1742  003062  012737  003142  020406        MOV     #VT1,RETURN     ;SET UP 'SCOPE' RETURN ADDRESS
1743  003070  012777  000340  176214        MOV     #340,@PSW       ;SET PROCESSOR PRIORITY TO '7'
1744  003076  012737  000001  024142        MOV     #1,TSTNUM       ;SET UP TEST NO.
1745  003104  005037  024432                CLR     SHFTSW
1746  003110  005037  024434                CLR     SHFPRT
1747  003114  005037  020404                CLR     SCOPEF
1748  003120  000207                        RTS     PC
1749
1750  003122  005737  024154        VTLOGIC:TST     TTYSWH          ;TTY AVAILABLE?
1751  003126  001403                        BEQ     .+10            ;NO
1752  003130  104012                        PRINT
1753  003132  023013                        MES20                   ;TEXT 'VT20 LOGIC TEST'
1754  003134  104022                        NULL
1755  003136  004737  003026        VTREST: JSR     PC,SETUPV       ;INITIALIZE THE LOGIC TEST
1756
1757                                ;*****************************************************************
1758                                ;TEST THAT THE 'CSR' WAS INITIALIZED CORRECTLY
1759                                ;*****************************************************************
1760
1761  003142  000240                VT1:    NOP                     ;'VT' TEST 1
1762  003144  000240                        NOP
1763  003146  012714  125212                MOV     #125212,@VTCSR4
1764  003152  000005                        RESET
1765  003154  011401                        MOV     @VTCSR4,R1      ;READ CONTENTS OF THE 'CSR'
1766  003156  042701  050000                BIC     #50000,R1       ;IGNORE 'BIT 12' AND 'BIT 14'
1767  003162  001401                        BEQ     .+4             ;RESET SHOULD HAVE CLEARED 'CSR'
1768  003164  104400                        ERROR                   ;'CSR' SHOULD = '0 OR 50000'
1769
1770                                ;*****************************************************************
1771                                ;TEST THAT THE 'MAINTENANCE' REGISTER WAS INITIALIZED TO '0'
1772                                ;*****************************************************************
1773
1774  003166  104001  000002        VT2:    SCOPE,2                 ;'VT' TEST 2
1775  003172  000005                        RESET
```

```
 1776  003174  011501                  MOV     @VTMAR5,R1      ;READ CONTENTS OF THE 'MAR'
 1777  003176  042701  050000          BIC     #50000,R1       ;IGNORE 'BIT 12' AND 'BIT 14'
 1778  003202  001401                  BEQ     .+4
 1779  003204  104400                  ERROR                   ;RESET DIDN'T CLR THE 'MAINT' REG. (EXCEPT BIT '12' AND
 1780                          ;*************************************************************************
 1781                          ;TEST FOR WRITING THE 'CSR' TO 125212
 1782                          ;*************************************************************************
 1783
 1784  003206  104001  000003  VT3:    SCOPE,3                 ;'VT' TEST 3
 1785  003212  012714  125212          MOV     #125212,@VTCSR4 ;WRITE 'CSR' TO '125212'
 1786  003216  011401                  MOV     @VTCSR4,R1      ;READ CONTENTS OF THE 'CSR'
 1787  003220  042701  050000          BIC     #50000,R1       ;IGNORE 'BIT 12' AND 'BIT 14'
 1788  003224  022701  125212          CMP     #125212,R1
 1789  003230  001401                  BEQ     .+4
 1790  003232  104400                  ERROR                   ;'CSR' SHOULD = '125212'
 1791  003234  005014                  CLR     @VTCSR4
 1792
 1793                          ;*************************************************************************
 1794                          ;TEST FOR WRITING THE 'CSR' TO '52504'
 1795                          ;*************************************************************************
 1796
 1797  003236  104001  000004  VT4:    SCOPE,4                 ;'VT' TEST 4
 1798  003242  012714  052504          MOV     #52504,@VTCSR4  ;WRITE 'CSR' TO '52504'
 1799  003246  104021                          WAITSYNC
 1800  003250  022714  052504          CMP     #52504,@VTCSR4
 1801  003254  001401                  BEQ     .+4             ;**BEWARE** BIT 6 IS CLEARED ON TYPEOUT
 1802  003256  104400                  ERROR                   ;'CSR' SHOULD = '52504'
 1803  003260  005014                  CLR     @VTCSR4         ;WRITE 'CSR' TO '0'
 1804
 1805                          ;*************************************************************************
 1806                          ;TEST FOR READING THE 'CSR' VIA THE 'MAINT' REG.
 1807                          ;*************************************************************************
 1808
 1809  003262  104001  000005  VT5:    SCOPE,5                 ;'VT' TEST 5
 1810  003266  012714  157760          MOV     #157760,@VTCSR4 ;SELECT 'MAINT CSR' VIA THE 'CSR'
 1811  003272  104021                          WAITSYNC
 1812  003274  022715  157760          CMP     #157760,@VTMAR5 ;MAINT SHOULD='CSR'
 1813  003300  001401                  BEQ     .+4             ;**BEWARE** BIT 6 IS CLEARED ON TYPEOUT
 1814  003302  104400                  ERROR                   ;MAINT REG. NOT ='CSR' (157760)
 1815  003304  005014                  CLR     @VTCSR4         ;CLR 'CSR'
 1816  003306  011501                  MOV     @VTMAR5,R1
 1817  003310  042701  050000          BIC     #50000,R1
 1818  003314  005701                  TST     R1              ;'MAR' SHOULD = 'CSR' (0)
 1819  003316  001401                  BEQ     .+4
 1820  003320  104400                  ERROR                   ;MAINT REG. NOT='CSR' (0)
 1821
 1822                          ;*************************************************************************
 1823                          ;WRITE THE 'CURSOR' ADDRESS REGISTER TO '125252'
 1824                          ;*************************************************************************
 1825
 1826  003322  104001  000006  VT6:    SCOPE,6                 ;'VT' TEST 6
 1827  003326  012777  125252  176150  MOV     #125252,@VTCAR  ;WRITE REG. TO ALTERNATE 1'S
 1828  003334  012777  000010  176146  MOV     #CAR,@VTCSR     ;SELECT 'CURSOR' ADDRESS VIA 'CSR'
 1829  003342  022715  125252          CMP     #125252,@VTMAR5 ;READ CURSOR ADDRESS VIA 'MAINT'
 1830  003346  001401                  BEQ     .+4
 1831  003350  104400                  ERROR                   ;CURSOR ADDRESS SHOULD=125252
```

```
1832  003352  005077  176132                  CLR     @VTCSR
1833  003356  005077  176122                  CLR     @VTCAR
1834
1835                                  ;*************************************************************************
1836                                  ;WRITE THE 'CURSOR' ADDRESS REGISTER TO '52525'
1837                                  ;*************************************************************************
1838
1839  003362  104001  000007          VT7:    SCOPE,7                         ;'VT' TEST 7
1840  003366  012777  052525  176110          MOV     #52525,@VTCAR           ;WRITE COMPLIMENT OF PART I
1841  003374  012777  000010  176106          MOV     #CAR,@VTCSR             ;SELECT 'CURSOR' ADDRESS VIA 'CSR'
1842  003402  022715  052525                  CMP     #52525,@VTMAR5          ;READ 'CAR' VIA 'MAR'
1843  003406  001401                          BEQ     .+4
1844  003410  104400                          ERROR                           ;'CAR' SHOULD=52525
1845  003412  005077  176072                  CLR     @VTCSR
1846  003416  005077  176062                  CLR     @VTCAR
1847
1848                                  ;*************************************************************************
1849                                  ;WRITE THE 'CURSOR' ADDRESS REGISTER TO '0'
1850                                  ;*************************************************************************
1851
1852  003422  104001  000010          VT10:   SCOPE,!0                                ;'VT' TEST 10
1853  003426  012777  177777  176050          MOV     #-1,@VTCAR              ;WRITE REG. TO -1
1854  003434  012714  000010                  MOV     #CAR,@VTCSR4            ;SELECT CURSOR ADDRESS
1855  003440  005077  176040                  CLR     @VTCAR                  ;WRITE REG. TO '0'
1856  003444  005715                          TST     @VTMAR5
1857  003446  001401                          BEQ     .+4
1858  003450  104400                          ERROR                           ;CURSOR ADDRESS SHOULD='0'
1859
1860                                  ;*************************************************************************
1861                                  ;TEST THAT THE 'CURSOR ADDRESS' REG. IS CLEARED VIA 'RESET'
1862                                  ;*************************************************************************
1863
1864  003452  104001  000011          VT11:   SCOPE,11                                ;'VT' TEST 11
1865  003456  012777  177777  176020          MOV     #-1,@VTCAR              ;WRITE REG. =-1
1866  003464  000005                          RESET                           ;ISSUE 'RESET' TO CLR REG.
1867  003466  012714  000010                  MOV     #CAR,@VTCSR4            ;SELECT 'CAR' VIA 'CSR'
1868  003472  005715                          TST     @VTMAR5                 ;TEST IF 'CAR' WAS CLEARED
1869  003474  001401                          BEQ     .+4
1870  003476  104400                          ERROR                           ;RESET DIDN'T CLR 'CAR' REG.
1871                                  ;*************************************************************************
1872                                  ;TEST FOR ROTATING A '1' THRU THE 'CAR'.
1873                                  ;*************************************************************************
1874
1875  003500  104001  000012          VT12:   SCOPE,12                                ;'VT' TEST 12
1876  003504  012703  000001                  MOV     #1,R3                   ;LOAD THE TEST REG. WITH '1'
1877  003510  010377  175770          LOOP12: MOV     R3,@VTCAR               ;LOAD THE 'CAR' WITH THE TEST REG.
1878  003514  012714  000010                  MOV     #CAR,@VTCSR4
1879  003520  020315                          CMP     R3,@VTMAR5
1880  003522  001401                          BEQ     .+4
1881  003524  104400                          ERROR                           ;'CAR' SHOULD = CONTAINS OF R3
1882  003526  005014                          CLR     @VTCSR4
1883  003530  005077  175750                  CLR     @VTCAR
1884  003534  006103                          ROL     R3                      ;ROTATE BIT
1885  003536  001364                          BNE     LOOP12
1886                                  ;*************************************************************************
1887                                  ;TEST FOR WRITING THE 'STARTING ADDRESS' REG. TO '125252'
```

```
1888                                    ;*************************************************************************
1889
1890  003540  104001  000013          VT13:   SCOPE,13                        ;'VT' TEST 13
1891  003544  012777  024474  175734          MOV     #VTBUFF,@VTSAR  ;LOAD LEGAL ADDR. TO PREVENT TIMEOUT
1892  003552  012714  000012                  MOV     #SAR,@VTCSR4    ;SELECT 'SAR' TO MAINT VIA 'CSR'
1893  003556  012777  125252  175722          MOV     #125252,@VTSAR  ;WRITE REG. TO ALTERNATE 1'S
1894  003564  104021                          WAITSYNC                ;WAIT FOR VERTICAL SYNC.
1895  003566  022715  125252                  CMP     #125252,@VTMAR5
1896  003572  001401                          BEQ     .+4
1897  003574  104400                          ERROR                   ;STARTING ADDR. REG. SHOULD=125252
1898
1899                                    ;*************************************************************************
1900                                    ;TEST FOR WRITING THE 'STARTING ADDRESS' REG. TO '52525'
1901                                    ;*************************************************************************
1902
1903  003576  104001  000014          VT14:   SCOPE,14                        ;'VT' TEST 14
1904  003602  012777  024474  175675          MOV     #VTBUFF,@VTSAR  ;LOAD LEGAL ADDR. TO PREVENT TIMEOUT
1905  003610  012714  000012                  MOV     #SAR,@VTCSR4    ;SELECT 'SAR' TO MAINT VIA 'CSR'
1906  003614  012777  052525  175664          MOV     #52525,@VTSAR   ;WRITE COMPLIMENT OF PART I
1907  003622  104021                          WAITSYNC                ;WAIT FOR VERTICAL SYNC.
1908  003624  022715  052525                  CMP     #52525,@VTMAR5
1909  003630  001401                          BEQ     .+4
1910  003632  104400                          ERROR                   ;'SAR' SHOULD =52525
1911                                    ;*************************************************************************
1912                                    ;TEST FOR WRITING THE 'STARTING ADDRESS' REG. TO '0'
1913                                    ;*************************************************************************
1914
1915  003634  104001  000015          VT15:   SCOPE,15                        ;'VT' TEST 15
1916  003640  012777  024474  175640          MOV     #VTBUFF,@VTSAR  ;LOAD LEGAL ADDR. TO PREVENT TIMEOUT
1917  003646  012714  000012                  MOV     #SAR,@VTCSR4    ;SELECT 'STARTING ADDR.' REG.
1918  003652  005077  175630                  CLR     @VTSAR          ;WRITE REG. TO '0'
1919  003656  104021                          WAITSYNC                ;WAIT FOR VERTICAL SYNC.
1920  003660  005715                          TST     @VTMAR5         ;CHECK FOR '0'
1921  003662  001401                          BEQ     .+4
1922  003664  104400                          ERROR                   ;STARTING ADDRESS REG. SHOULD='0'
1923                                    ;*************************************************************************
1924                                    ;TEST FOR ROTATING A '1' THUR THE STARTING ADDRESS REG.
1925                                    ;NOTE: 'R3' IS USED AS A CHECKING ADDRESS AND WILL CONTAIN THE
1926                                    ;EXPECTED DATA ON A FAILURE.
1927                                    ;*************************************************************************
1928
1929  003666  104001  000016          VT16:   SCOPE,16                        ;'VT' TEST 16
1930  003672  012777  024474  175606          MOV     #VTBUFF,@VTSAR  ;LOAD LEGAL ADDR. TO PREVENT TIMEOUT
1931  003700  012703  000001                  MOV     #1,R3           ;LOAD THE TEST REG. WITH '1'
1932  003704  012714  000012          LOOP16: MOV     #SAR,@VTCSR4    ;SELECT 'STARTING ADDR.' REG.
1933  003710  010377  175572                  MOV     R3,@VTSAR       ;LOAD 'SAR' FROM 'R3'
1934  003714  104021                          WAITSYNC                ;WAIT FOR VERTICAL SYNC.
1935  003716  020315                          CMP     R3,@VTMAR5      ;CHECK IF REG. LOADED CORRECTLY.
1936  003720  001401                          BEQ     .+4
1937  003722  104400                          ERROR                   ;'SAR SHOULD = CONTAINS OF 'R3'
1938  003724  006103                          ROL     R3              ;ROTATE BIT
1939  003726  001366                          BNE     LOOP16          ;LOOP  UNTIL CARRY IS SET
1940                                    ;*************************************************************************
1941                                    ;TEST THAT 'RESET' WILL CLR THE 'STARTING ADDRESS' REG.
1942                                    ;*************************************************************************
1943
```

```
1944  003730  104001  000017          VT17:   SCOPE,17                        ;'VT' TEST 17
1945  003734  012777  024474  175544           MOV     #VTBUFF,@VTSAR  ;LOAD LEGAL ADDR. TO PREVENT TIMEOUT
1946  003742  012714  000012                   MOV     #SAR,@VTCSR4    ;SELECT 'STARTING ADDRESS' REG.
1947  003746  012777  177777  175532           MOV     #-1,@VTSAR      ;WRITE REG. TO '-1'
1948  003754  000005                           RESET                   ;RESET SHOULD CLR REG.
1949  003756  104021                           WAITSYNC                ;WAIT FOR VERTICAL SYNC.
1950  003760  012714  000012                   MOV     #SAR,@VTCSR4    ;RE-SELECT THE 'SAR'
1951  003764  005715                           TST     @VTMAR5
1952  003766  001401                           BEQ     .+4
1953  003770  104400                           ERROR                   ;RESET DIDN'T CLR 'SAR'
1954
1955                                  ;*************************************************************************
1956                                  ;TEST THAT THE 'DAR' (DATA ADDRESS REG.) IS LOADED FROM 'SAR' (START ADDR. REG.)
1957                                  ;*************************************************************************
1958
1959  003772  104001  000020          VT20:   SCOPE,20                        ;'VT' TEST 20
1960  003776  012777  125252  175502           MOV     #125252,@VTSAR  ;LOAD THE 'SAR'
1961  004004  104021                           WAITSYNC                ;WAIT FOR VERTICAL SYNC.
1962  004006  012714  000002                   MOV     #DAR,@VTCSR4    ;SELECT MAINT. 'DAR' VIA 'CSR'
1963  004012  022715  125252                   CMP     #125252,@VTMAR5
1964  004016  001401                           BEQ     .+4
1965  004020  104400                           ERROR                   ;'DAR' WASN'T LOADED FROM 'SAR'
1966  004022  005014                           CLR     @VTCSR4
1967
1968                                  ;*************************************************************************
1969                                  ;ISSUE RESET TO CLR TIME-OUT FROM PREVIOUS TEST
1970                                  ;*************************************************************************
1971
1972  004024  104001  000021          VT21:   SCOPE,21                        ;'VT' TEST 21
1973  004030  000005                           RESET
1974
1975                                  ;*************************************************************************
1976                                  ;TEST THAT THE 'DAR' (DATA ADDRESS REG.) IS LOADED FROM 'SAR' (START ADDR. REG.)
1977                                  ;*************************************************************************
1978
1979  004032  104001  000022          VT22:   SCOPE,22                        ;'VT' TEST 22
1980  004036  012777  052525  175442           MOV     #052525,@VTSAR  ;LOAD THE 'SAR'
1981  004044  104021                           WAITSYNC                ;WAIT FOR VERTICAL SYNC.
1982  004046  012714  000002                   MOV     #DAR,@VTCSR4    ;SELECT MAINT. 'DAR'
1983  004052  022715  052524                   CMP     #052524,@VTMAR5
1984  004056  001401                           BEQ     .+4
1985  004060  104400                           ERROR                   ;'DAR' WASN'T LOADED FROM 'SAR'
1986  004062  005014                           CLR     @VTCSR4
1987                                  ;*************************************************************************
1988                                  ;ISSUE RESET TO CLR TIME-OUT FROM PREVIOUS TEST
1989                                  ;*************************************************************************
1990
1991  004064  104001  000023          VT23:   SCOPE,23                        ;'VT' TEST 23
1992  004070  000005                           RESET
1993                                  ;*************************************************************************
1994                                  ;TEST FOR ROTATING A '1' THUR THE 'DAR' (DAR IS LOADED FROM THE SAR).
1995                                  ;*************************************************************************
1996
1997  004072  104001  000024          VT24:   SCOPE,24                        ;'VT' TEST 24
1998  004076  012703  000002                   MOV     #2,R3           ;LOAD THE TEST REG. WITH '2'
1999  004102  010377  175400          LOOP24: MOV     R3,@VTSAR       ;LOAD THE 'SAR'
```

```
2000  004106  104021                          WAITSYNC                ;WAIT FOR VERTICAL SYNC.
2001  004110  012714  000002                  MOV     #DAR,@VTCSR4    ;SELECT MAINT. 'DAR' VIA 'CSR'
2002  004114  020315                          CMP     R3,@VTMAR5      ;TEST 'DAR'
2003  004116  001401                          BEQ     .+4
2004  004120  104400                          ERROR                   ;'DAR' WASN'T LOADED FROM 'SAR'
2005  004122  005014                          CLR     @VTCSR4
2006  004124  006103                          ROL     R3              ;ROTATE BIT
2007  004126  001365                          BNE     LOOP24          ;ROTATE THRU REG.
2008
2009                                  ;*************************************************************************
2010                                  ;ISSUE RESET TO CLR TIME-OUT FROM PREVIOUS TEST
2011                                  ;*************************************************************************
2012
2013  004130  104001  000025          VT25:   SCOPE,25                        ;'VT' TEST 25
2014  004134  000005                          RESET
2015
2016                                  ;*************************************************************************
2017                                  ;TEST THAT THE 'CURSOR ADDR.' BIT IS SET IN THE 'CSR' IF THE 'CAR'
2018                                  ;AND THE 'SAR' ARE SET EQUAL.  ALSO BIT 14 CLEARS IF CAR=SAR
2019                                  ;*************************************************************************
2020
2021  004136  104001  000026          VT26:   SCOPE,26                        ;'VT' TEST 26
2022  004142  012777  020727  175336          MOV     #SHFTBF+1,@VTSAR        ;LOAD 'SAR'
2023  004150  104021                          WAITSYNC                        ;WAIT FOR VERTICAL SYNC.
2024  004152  012777  020727  175324          MOV     #SHFTBF+1,@VTCAR        ;LOAD 'CAR' = 'SAR'
2025  004160  012714  000004                  MOV     #SRIDE,@VTCSR4          ;FORCE 'NPR' TRANSFER
2026  004164  032714  010000                  BIT     #10000,@VTCSR4          ;TEST IF BIT 12 (CAR BIT) SET
2027  004170  001001                          BNE     .+4
2028  004172  104400                          ERROR                           ;'CAR' BIT IN 'CSR' DIDN'T SET W/'CAR' = 'SAR'
2029  004174  032714  040000                  BIT     #40000,@VTCSR4          ;TEST IF BIT14 CLEAR
2030  004200  001401                          BEQ     .+4                     ;SHOULD BE WHEN SAR=CAR
2031  004202  104400                          ERROR                           ;BIT 14 NOT CLEAR
2032  004204  005014                          CLR     @VTCSR4
2033                                  ;*************************************************************************
2034                                  ;TEST THAT THE 'CURSOR ADDR.' BIT IS CLEARED IF THE 'CAR' AND 'SAR'
2035                                  ;ARE NOT SET EQUAL.
2036                                  ;*************************************************************************
2037
2038  004206  104001  000027          VT27:   SCOPE,27                        ;'VT' TEST 27
2039  004212  012777  020741  175266          MOV     #SHFTBF+13,@VTSAR       ;LOAD 'SAR'
2040  004220  104021                          WAITSYNC                        ;WAIT FOR VERTICAL SYNC.
2041  004222  012777  020737  175254          MOV     #SHFTBF+11,@VTCAR       ;LOAD 'CAR' = 'SAR'
2042  004230  012714  000004                  MOV     #SRIDE,@VTCSR4  ;FORCE 'NPR' TRANSFER
2043  004234  032715  010000                  BIT     #10000,@VTMAR5          ;TEST IF BIT 12 (CAR BIT) SET
2044  004240  001401                          BEQ     .+4
2045  004242  104400                          ERROR                           ;'CAR' SET W/ 'CAR' NOT = 'SAR'
2046  004244  032777  004000  175236          BIT     #4000,@VTCSR    ;SEE IF BIT 14 IN CSR, SET SHOULD BE
2047  004252  001001                          BNE     .+4             ;WHEN CAR <> SAR WITH ECO FOR BIT 14
2048  004254  000240                          NOP                     ;NOP IF NO ECO, PUT IN 104400 IN THIS
2049                                                                  ;LOCATION IF ECO IS IN.
2050  004256  005014                          CLR     @VTCSR4
2051                                  ;*************************************************************************
2052                                  ;TEST THE 'EOS' (END OF SCREEN) BIT IS SET IF THE 'EOS' CHAR. IS TRANSFERRED
2053                                  ;*************************************************************************
2054
2055  004260  104001  000030          VT30:   SCOPE,30                        ;'VT' TEST 30
```

```
2056  004264  012777  021024  175214          MOV     #SHFEND,@VTSAR          ;LOAD 'SAR'
2057  004272  104021                          WAITSYNC                        ;WAIT FOR VERTICAL SYNC.
2058  004274  012714  000004                  MOV     #SRIDE,@VTCSR4          ;FORCE 'NPR' TRANSFER
2059  004300  032714  020000                  BIT     #20000,@VTCSR4          ;TEST IF BIT 13 (EOS)
2060  004304  001001                          BNE     .+4
2061  004306  104400                          ERROR                           ;'EOS' DIDN'T SET IN 'CSR'
2062  004310  005014                          CLR     @VTCSR4
2063
2064                                  ;*************************************************************************
2065                                  ;TEST THE 'SHIFT REG. INPUT DATA EVEN' GETS TRANSFERRED FROM THE DATA BUFFER
2066                                  ;TO THE 'MAR'.  A SERIES OF '64' SHIFTS ARE DONE AND THE 'MAR' IS COMPARED
2067                                  ;TO THE DATA BUFFER AFTER EACH SHIFT.
2068                                  ;*************************************************************************
2069
2070  004312  104001  000031          VT31:   SCOPE,31                        ;'VT' TEST 31
2071  004316  000005                          RESET
2072  004320  012737  000010  020402          MOV     #10,ICOUNT              ;RESET ITERATION COUNT TO '8'
2073  004326  012737  023544  020026          MOV     #MES37,SHFMES           ;SET UP TO PRINT ON ERROR
2074  004334  012737  000004  024430          MOV     #SRIDE,SELECT           ;SELECT 'SHIFT INPUT DATA EVEN'
2075  004342  004737  004752                  JSR     PC,INITSF               ;INITIALIZE SHIFT TEST
2076  004346  004737  005004                  JSR     PC,SHIFT                ;SUBROUTINE TO FORCE '64' 'NPR' XFER
2077  004352  104400                          ERROR                           ;'SHIFT INPUT DATA EVEN' SHIFT ERROR
2078  004354  000415                          BR      TAGVB                   ;EXIT ON ERROR
2079                                  ;*************************************************************************
2080                                  ;AT THIS POINT '64' SHIFTS HAVE BEEN MADE TO THE 'SHIFT REG. INPUT DATA
2081                                  ;EVEN'. THE NEXT TEST EXECUTES '63' MORE SHIFTS VALIDATING THE
2082                                  ;'SHIFT REG. OUTPUT DATA EVEN' INPUTTED FROM THE PREVIOUS TEST.
2083                                  ;*************************************************************************
2084
2085  004356  012737  023555  020026          MOV     #MES38,SHFMES
2086  004364  012737  000006  024430          MOV     #SRODE,SELECT   ;SELECT 'SHIFT OUTPUT DATA EVEN'
2087  004372  004737  004752                  JSR     PC,INITSF       ;INITIALIZE SHIFT TEST
2088  004376  005722                          TST     (R2)+           ;UPDATE BUFFER COMPARE POINTER BY '2'
2089  004400  005200                          INC     R0              ;SHIFT ONLY 63 TIMES FOR OUTPUT
2090  004402  004737  005004                  JSR     PC,SHIFT        ;SUBROUTINE TO FORCE THE '63' NPR XFERS
2091  004406  104400                          ERROR                   ;'SHIFT OUTPUT DATA EVEN' SHIFT ERROR
2092  004410  000240                  TAGVB:  NOP
2093                                  ;*************************************************************************
2094                                  ;TEST THE 'SHIFT REG. INPUT DATA ODD' GETS TRANSFERRED FROM THE DATA BUFFER
2095                                  ;TO THE 'MAR'.  A SERIES OF '64' SHIFTS ARE DONE AND THE 'MAR' IS COMPARED
2096                                  ;TO THE DATA BUFFER AFTER EACH SHIFT.
2097                                  ;*************************************************************************
2098
2099  004412  104001  000032          VT32:   SCOPE,32                        ;'VT' TEST 32
2100  004416  000005                          RESET
2101  004420  012737  023544  020026          MOV     #MES37,SHFMES
2102  004426  012737  000014  024430          MOV     #SRIDO,SELECT   ;SELECT 'SHIFT INPUT DATA ODD'
2103  004434  004737  004752                  JSR     PC,INITSF       ;INITIALIZE SHIFT TEST
2104  004440  004737  005004                  JSR     PC,SHIFT        ;SUBROUTINE TO FORCE '1' 'NPR' XFER
2105  004444  104400                          ERROR                   ;'SHIFT INPUT DATA ODD' SHIFT ERROR
2106  004446  000415                          BR      TAGVC           ;EXIT ON ERROR
2107
2108                                  ;*************************************************************************
2109                                  ;AT THIS POINT '64' SHIFTS HAVE BEEN MADE TO THE 'SHIFT REG. INPUT DATA
2110                                  ;ODD'. THIS NEXT TEST EXECUTES '63' MORE SHIFTS VALIDATING THE
2111                                  ;'SHIFT REG. OUTPUT DATA ODD' INPUTTED FROM THE PREVIOUS TEST.
```

```
2112                                    ;*************************************************************************
2113
2114  004450  012737  023555  020026            MOV     #MES38,SHFMES
2115  004456  012737  000016  024430            MOV     #SR000,SELECT   ;SELECT 'SHIFT OUTPUT DATA ODD'
2116  004464  004737  004752                    JSR     PC,INITSF       ;INITIALIZE SHIFT TEST
2117  004470  005722                            TST     (R2)+           ;UPDATE BUFFER COMPARE POINTER BY '2'
2118  004472  005200                            INC     R0              ;SHIFT ONLY 63 TIMES FOR OUTPUT
2119  004474  004737  005004                    JSR     PC,SHIFT        ;SUBROUTINE TO FORCE THE '64' NPR XFERS
2120  004500  104400                            ERROR                   ;'SHIFT OUTPUT DATA ODD' SHIFT ERROR
2121  004502  000240                    TAGVC:  NOP
2122                                    ;*************************************************************************
2123                                    ;TEST 'NXM' FOR CAUSING AN INTERRUPT VIA LOADING A 'NXM' ADDRESS IN THE 'SAR'
2124                                    ;NOTE: THIS IS THE FIRST TIME THAT THE 'INTR. ENABLE BIT IS SET OR USED.
2125                                    ;*************************************************************************
2126
2127  004504  104001  000033            VT33:   SCOPE,33                        ;'VT' TEST 33
2128  004510  012737  000100  020402            MOV     #100,ICOUNT             ;RESET ITERATION COUNT TO '100'
2129  004516  005037  024432                    CLR     SHFTSW          ;CLR SOFTWARE SW.
2130  004522  012777  173000  174756            MOV     #173000,@VTSAR  ;LOAD 'SAR' W/ 'NXM' ADDRESS
2131  004530  004737  016740                    JSR     PC,LDVTVT       ;LOAD THE 'VT' VECTOR ADDRESS
2132  004534  004544                            TAGVA
2133  004536  104021                            WAITSYNC                ;WAIT FOR INTERRUPT
2134  004540  104021                            WAITSYNC
2135  004542  104400                            ERROR                   ;SETTING 'SAR' TO 'NXM' ADDR. DIDN'T ERROR
2136  004544  013706  024132            TAGVA:  MOV     STACK,SP        ;RESET STACK POINTER
2137  004550  004737  016774                    JSR     PC,CLRVTV       ;RESET 'VT' VECTOR ADDR. TO HALT
2138  004554  005714                            TST     @VTCSR4         ;TEST THE 'NXM' SET THE ERROR BIT
2139  004556  100401                            BMI     .+4
2140  004560  104400                            ERROR                   ;'NXM' FAILED TO SET 'ERROR' BIT
2141  004562  000005                            RESET
2142
2143                                    ;*************************************************************************
2144                                    ;TEST THAT NO INTERRUPT OCCURS WITH PROC. AT PRIORITY '5'
2145                                    ;*************************************************************************
2146
2147  004564  104001  000034            VT34:   SCOPE,34                        ;'VT' TEST 34
2148  004570  012777  173000  174710            MOV     #173000,@VTSAR
2149  004576  004737  016740                    JSR     PC,LDVTVT       ;LOAD 'VT' INTERRUPT VECTOR
2150  004602  004624                            TAGVD
2151  004604  012777  000240  174500            MOV     #240,@PSW
2152  004612  012700  170000                    MOV     #170000,R0
2153  004616  005200                            INC     R0
2154  004620  001376                            BNE     .-2
2155  004622  000403                            BR      .+10            ;'OK', NO INTERRUPT OCCURRED
2156  004624  104400                    TAGVD:  ERROR                   ;INTERRUPTED WITH PROC @ PRIOR '5'
2157  004626  013706  024132                    MOV     STACK,SP
2158  004632  000005                            RESET
2159  004634  004737  016774                    JSR     PC,CLRVTV
2160                                    ;*************************************************************************
2161                                    ;TEST THAT NO 'NXM' ERRORS OCCUR IF THE 'VT' IS RUNNING AND THE
2162                                    ;STARTING ADDRESS REG. IS CHANGED.
2163                                    ;*************************************************************************
2164
2165  004640  104001  000035            VT35:   SCOPE,35                        ;'VT' TEST 35
2166  004644  012701  000010                    MOV     #10,R1          ;SET UP A LOOP COUNTER
2167  004650  012700  020730                    MOV     #SHFTBF+2,R0
```

```
2168  004654  012777  020726  174624          MOV     #SHFTBF,@VTSAR
2169  004662  104021                          WAITSYNC                ;WAIT FOR VERICAL SYNC
2170  004664  004737  016740                  JSR     PC,LDVTVT       ;LOAD 'VT' INTERRUPT VECTOR ADDR.
2171  004670  004716                          TAGVF
2172  004672  005002                  TAGVE:  CLR     R2
2173  004674  005202                          INC     R2              ;EVERYTIME 'R2' OVERFLOWS THE
2174  004676  001376                          BNE     .-2             ;'SAR' IS RELOADED
2175  004700  010077  174602                  MOV     R0,@VTSAR       ;RELOAD 'SAR'
2176  004704  005720                          TST     (R0)+
2177  004706  104021                          WAITSYNC                ;WAIT FOR VERICAL SYNC
2178  004710  005301                          DEC     R1
2179  004712  001367                          BNE     TAGVE
2180  004714  000403                          BR      .+10            ;OK, NO INTERRUPTS OCCURRED
2181  004716  104400                  TAGVF:  ERROR                   ;'NXM' ERROR OCCURRED WHEN 'SAR' WAS CHANGED
2182  004720  013706  024132                  MOV     STACK,SP
2183  004724  004737  016774                  JSR     PC,CLRVTV
2184
2185                                  ;*************************************************************************
2186                                  ;'VT' LOGIC TEST COMPLETE
2187                                  ;*************************************************************************
2188
2189  004730  104001  000036          VT36:   SCOPE,36                        ;'VT' TEST 36
2190  004734  104012                          PRINT
2191  004736  021610                          MES7                    ;TEXT 'TEST COMPLETE'
2192  004740  104006                          ENDTST                  ;END OF TEST
2193  004742  000401                          BR      .+4
2194  004744  000510                          BR      VTCHAR          ;CONTINUE TO CHARACTER TEST
2195  004746  000137  003136                  JMP     VTREST          ;RESTART THE LOGIC TEST.
2196
2197                                  ;*************************************************************************
2198                                  ;SUBROUTINE TO INITIALIZE THE 'SAR', 'CAR' AND SHIFT COUNTER FOR THE
2199                                  ;'VT' LOGIC SHIFT TEST.
2200                                  ;*************************************************************************
2201
2202  004752  012777  020726  174526  INITSF: MOV     #SHFTBF,@VTSAR  ;LOAD 'STARTING ADDR'
2203  004760  012703  020726                  MOV     #SHFTBF,R3
2204  004764  104021                          WAITSYNC                ;WAIT FOR VERTICAL SYNC.
2205  004766  012702  020726                  MOV     #SHFTBF,R2
2206  004772  005000                          CLR     R0              ;CLR 'SHIFT' COUNTER
2207  004774  012737  000001  024432          MOV     #1,SHFTSW       ;SET SOFTWARE SW.
2208  005002  000207                          RTS     PC
2209                                  ;*************************************************************************
2210                                  ;SUBROUTINE TO FORCE THE 'NPR' XFER FOR THE 'VT' LOGIC SHIFT TEST.
2211                                  ;THE ADDRESS 'SELECT' CONTAINS THE SELECTED SHIFT FUNCTION (SHIFT INPUT
2212                                  ;DATA EVEN, ETC.) WHICH IS EXECUTED.
2213                                  ;*************************************************************************
2214
2215  005004  005200                  SHIFT:  INC     R0              ;INCREMENT SOFTWARE SHIFT COUNTER
2216  005006  013714  024430                  MOV     SELECT,@VTCSR4  ;FORCE 'NPR' TRANSFER
2217  005012  005723                          TST     (R3)+           ;UPDATE STARTING ADDRESS BY '2'
2218  005014  010377  174466                  MOV     R3,@VTSAR       ;SET UP TO GET NEXT BUFFER WORD.
2219  005020  104021                          WAITSYNC                ;WAIT FOR VERTICAL SYNC.
2220  005022  012201                          MOV     (R2)+,R1        ;GET XFER'D WORD FROM DATA BUFFER
2221  005024  042701  000360                  BIC     #360,R1         ;CLR UNWANTED BITS FROM CONTROL WORD
2222  005030  000301                          SWAB    R1              ;SWAP BYTES TO COMPARE TO 'MAR'
2223  005032  022700  000040          SHFLST: CMP     #32.,R0         ;IS THIS THE '32' SHIFT?
```

```
2224  005036  001010                        BNE     SHF64           ;NO, CHECK FOR '64' SHIFT
2225  005040  022715  020031                CMP     #20031,@VTMAR5  ;YES,END OF SCREEN SHOULD BE SET IN 'MAR'
2226  005044  001013                        BNE     SHFBAD          ;NO, 32ND SHIFT IS BAD
2227  005046  004737  004752        SHF32:  JSR     PC,INITSF       ;RE-INITIALIZE SOFWARE POINTERS FOR '32' MORE SHIFTS
2228  005052  012700  000040                MOV     #32.,R0         ;RESET SHIFT COUNTER TO COUNT TO '64'
2229  005056  000752                        BR      SHIFT           ;DO IT.
2230  005060  022700  000100        SHF64:  CMP     #64.,R0         ;IS THIS THE LAST SHIFT
2231  005064  001006                        BNE     TSTSHF          ;NO, COMPARE DATA TO 'MAR' DIRECTLY
2232  005066  022715  020031                CMP     #20031,@VTMAR5  ;YES, 'EOS' SHOULD BE SET
2233  005072  001432                        BEQ     EXITSF          ;EXIT
2234  005074  052701  020000        SHFBAD: BIS     #20000,R1       ;SET UP 'R1' TO = EXPECTED WORD
2235  005100  000412                        BR      REPORT          ;REPORT ERROR
2236  005102  032701  000400        TSTSHF: BIT     #400,R1         ;COMPLIMENT BLANK BIT FOR COMPARE
2237  005106  001003                        BNE     .+10
2238  005110  052701  000400                BIS     #400,R1
2239  005114  000402                        BR      .+6
2240  005116  042701  000400                BIC     #400,R1
2241  005122  020115                        CMP     R1,@VTMAR5      ;DOES 'MAR' = EXPT'D WORD?
2242  005124  001727                        BEQ     SHIFT           ;YES, FORCE NEXT SHIFT
2243  005126  022700  000100        REPORT: CMP     #64.,R0         ;IS ERROR ON THE LAST SHIFT?
2244  005132  001404                        BEQ     .+12            ;YES,EXIT
2245  005134  032777  010000  174162        BIT     #SW12,@SWR      ;NO, IS SW SET TO CONTINUE SHIFTING?
2246  005142  001401                        BEQ     .+4             ;YES
2247  005144  000207                        RTS     PC              ;RETURN TO SUBTEST AND REPORT ERROR
2248  005146  104400                        ERROR                   ;PRINT ERROR DATA
2249  005150  022700  000040                CMP     #32.,R0         ;IS THIS THE '32' SHIFT
2250  005154  001734                        BEQ     SHF32           ;YES, RE-INITIALIZE BUFFER
2251  005156  000712                        BR      SHIFT           ;NO,CONTINUE SHIFTING
2252  005160  062716  000004        EXITSF: ADD     #4,(SP)         ;SET UP STACK TO SKIP OVER ERROR
2253  005164  000207                        RTS     PC
2254                          ;*****************************************************************
2255                          ;VT 'CHARACTER DISPLAY' TEST (1)
2256                          ;THIS TEST DISPLAYS ALL '160' VT CHARACTERS ONE AT A TIME UNTIL THE ENTIRE
2257                          ;'VT' SCREEN IS FILLED. EACH CHARACTER IS DISPLAYED FOR APPROXIMATELY '3' SECONDS
2258                          ;BEFORE THE NEXT CHARACTER IS DISPLAYED.  ON EACH LINE,THE '64' CHARACTER IS DIS-
2259                          ;PLAYED AS EITHER A 'VISIBLE END OF LINE OR A 'VISIBLE END OF PARAGRAPH'.
2260                          ;DATA 'SW14' MAY BE SET AT ANYTIME TO HOLD ON ANY INDIVIDUAL CHARACTER.
2261                          ;*****************************************************************
2262
2263  005166  012737  005166  020406  VTCHAR: MOV     #VTCHAR,RETURN
2264  005174  104000                        DISPLAY
2265  005176  023204                        MES24                           ;TEXT "VT CHARACTER DISPLAY TEST".
2266  005200  104005                        EOSBUF                          ;LOAD 'VT' BUFFER WITH 'EOS'.
2267  005202  005037  024152                CLR     INTSWH                  ;SUBROUTINE SOFTWARE SW.
2268  005206  000401                        BR      .+4
2269  005210  104004                        DELAY                           ;3 SEC. DELAY.
2270  005212  004737  015170                JSR     PC,INTCHR               ;CREATE & DISPLAY NEXT CHAR.
2271  005216  000240                        NOP                             ;NORMAL RETURN TO HERE
2272  005220  000773                        BR      .-10                    ;END OF LINE RETURNS HERE
2273  005222  104006                        ENDTST
2274  005224  000760                        BR      VTCHAR                  ;END OF SCREEN RETURN HERE
2275
2276                          ;*****************************************************************
2277                          ;VT 'FIELD CONTROL' TEST (2)
2278                          ;THIS TEST DISPLAYS THE 'VT' CHARACTER SET USING ALL '5' FIELD CONTROL MODES.
2279                          ;(NORMAL,BLINK,BOLD,BLANK AND UNDERLINE). THE TEST DISPLAYS CHARACTERS
```

```
                                ;'40-136' ON LINES '1-5' USING ALL '5' MODES. CHARACTERS '137-240 (WITH THE
                                ;EXCEPTION OF CHARACTERS 212,214 & 231) ON LINES '6-10'. CHARACTERS
                                ;'241-277' ARE DISPLAYED ON THE LAST '5' LINES.
                                ;*****************************************************************
005226  012737  005226  020406  VTMOD1: MOV     #VTMOD1,RETURN
005234  104000                          DISPLAY
005236  023240                          MES25                           ;TEXT 'VT FIELD CONTROL TEST'.
005240  104005                          EOSBUF                          ;LOAD 'VT' BUFFER WITH 'VISIBLE EOS'.
005242  012737  024474  024460          MOV     #VTBUFF,KSTOR1
005250  005037  024152                  CLR     INTSWH                  ;CLR SOFTWARE SWITCH
005254  012737  177775  024464          MOV     #-3,KSTOR3              ;SET UP FOR '3' SETS OF CHAR.'S
005262  004737  005274                  JSR     PC,VTMD1A               ;LOAD CHAR.'S
005266  104023                          THEND
005270  000756                          BR      VTMOD1
005272  000413                          BR      VTMOD2                  ;GO TO NEXT TEST IF SW. IS SET

005274  004737  015170          VTMD1A: JSR     PC,INTCHR               ;CREATE AND LOAD CHAR.
005300  000775                          BR      .-4                     ;LOOP UNTIL LINE IS FULL
005302  000240                          NOP                             ;END OF LINE RETURNS HERE
005304  104007                          SETLNE                          ;SET UP '4' LINES USING ALL MODES
005306  010037  024460                  MOV     R0,KSTOR1               ;SAVE ST. ADDR. FOR NEXT BUFFERED LINE
005312  005237  024464                  INC     KSTOR3                  ;FINISHED?
005316  001366                          BNE     VTMD1A                  ;NO, CREATE NEXT SET
005320  000207                          RTS     PC                      ;YES, EXIT
                                ;*****************************************************************
                                ;VT 'CSR PRESET' TEST (3)
                                ;THIS TEST DISPLAYS THE 'VT' CHARACTER SET IN ALL FIELD CONTROL MODES
                                ;(BOLD,BLINK,BLANK & UNDERLINE). THE FIELD CONTROL IS SELECTED VIA THE
                                ;'CSR' BEFORE EACH CHARACTER SET IS DISPLAYED.
                                ;*****************************************************************

005322  012737  005322  020406  VTMOD2: MOV     #VTMOD2,RETURN
005330  104000                          DISPLAY
005332  023270                          MES26                           ;TEXT 'VT' CSR PRESET TEST'.
005334  104005                          EOSBUF                          ;LOAD 'VT' BUFFER W/ 'EOS'.
005336  104000                          DISPLAY
005340  023566                          MES39                           ;TEXT 'BOLD PRESET TEST'
005342  012777  002001  174140  VTMD2A: MOV     #2001,@VTCSR            ;SELECT 'BOLD' PRESET & GO
005350  004737  005434                  JSR     PC,VTMD2B               ;LOAD CHAR.'S
005354  104000                          DISPLAY
005356  023611                          MES40                           ;TEXT 'BLINK PRESET TEST'
005360  012777  001001  174122          MOV     #1001,@VTCSR             ;SELECT 'BLINK' PRESET
005366  004737  005434                  JSR     PC,VTMD2B               ;LOAD CHAR.'S
005372  104000                          DISPLAY
005374  023635                          MES41                           ;TEXT 'BLANK PRESET TEST'
005376  012777  000401  174104          MOV     #401,@VTCSR             ;SELECT 'BLANK' PRESET
005404  004737  005434                  JSR     PC,VTMD2B               ;LOAD CHAR.'S
005410  104000                          DISPLAY
005412  023661                          MES42                           ;TEXT 'UNDERLINE PRESET TEST'
005414  012777  004001  174066          MOV     #4001,@VTCSR            ;SELECT 'UNDERLINE' PRESET
005422  004737  005434                  JSR     PC,VTMD2B               ;LOAD CHAR.'S
005426  104006                          ENDTST
005430  000734                          BR      VTMOD2                  ;RE-CYCLE
005432  000410                          BR      VTEOL                   ;BRANCH TO NEXT TEST IS SW. SET

```

```
2336  005434  005037  024152          VTMD2B: CLR     INTSWH                  ;CLR SOFTWARE SW.
2337  005440  004737  015170                  JSR     PC,INTCHR               ;CREATE & DISPLAY CHAR.
2338  005444  000240                          NOP
2339  005446  000774                          BR      .-6                     ;LOOP UNTIL SCREEN IF FILLED
2340  005450  104020                          DELAYL                          ;6 SEC. DELAY
2341  005452  000207                          RTS     PC                      ;EXIT
2342                                  ;*************************************************************************
2343                                  ;VT 'END OF LINE' TEST (4)
2344                                  ;THIS TEST TESTS THAT THE 'END OF LINE' CHARACTER CAN BE PLACED IN
2345                                  ;ANY OF THE '1024' CHARACTER POSITIONS ON THE SCREEN. THIS IS DONE BY
2346                                  ;STARTING ALL 16 LINES WITH A 'VISIBLE END OF LINE' CHARACTER AND
2347                                  ;THEN INCREMENTING THESE CHARACTERS ACROSS THE SCREEN.
2348                                  ;*************************************************************************
2349
2350  005454  012737  005454  020406  VTEOL:  MOV     #VTEOL, RETURN
2351  005462  104000                          DISPLAY
2352  005464  023315                          MES27                           ;TEXT 'VT END OF LINE TEST'
2353  005466  104005                          EOSBUF                          ;LOAD 'VT' BUFFER & START
2354  005470  005001                          CLR     R1                      ;CLR LINE CNTR.
2355  005472  112720  000212          RESCNT: MOVB    #VISEOL,(R0)+           ;TERMINATE LINE W/ 'EOL'
2356  005476  012703  000077                  MOV     #63.,R3                 ;SAVE '63' BYTES BEFORE NEXT LINE
2357  005502  105020                          CLRB    (R0)+
2358  005504  005303                          DEC     R3
2359  005506  001375                          BNE     .-4
2360  005510  005201                          INC     R1                      ;INCREMENT LINE CNTR.
2361  005512  022701  000021                  CMP     #17.,R1                 ;DONE '16' LINES ?
2362  005516  001365                          BNE     RESCNT                  ;NO, CREATE NEXT LINE
2363  005520  104004                          DELAY                           ;DISPLAY THE 1ST 16 LINES OF 'EOL'
2364
2365                                  ;AT THIS POINT 16 LINES EACH CONTAINING 1 'EOL' AND 63 SPACES HAVE
2366                                  ;BEEN SET UP IN MEMORY. THE FOLLOWING SUBROUTINE GOES THRU ONE
2367                                  ;BY ONE REPLACING THE 'EOL' WITH AN '*' AND MOVING THE 'EOL'
2368                                  ;INTO THE NEXT SPACE UNTIL THE ENTIRE LINE HAS BEEN CHECKED.
2369
2370  005522  012705  000077                  MOV     #63.,R5                 ;SET UP FOR '63' SHIFTS
2371  005526  012704  000020          SRHEOL: MOV     #16.,R4
2372  005532  012700  024474                  MOV     #VTBUFF,R0              ;RESET BUFFER ADDR. POINTER
2373  005536  122027  000212          SHFEOL: CMPB    (R0)+,#VISEOL           ;SEARCH BUFFER FOR "EOL"
2374  005542  001375                          BNE     .-4
2375  005544  112740  000052                  MOVB    #52,-(R0)               ;REPLACE 'EOL' W/ '*'
2376  005550  105720                          TSTB    (R0)+                   ;RESET POINTER
2377  005552  112720  000212                  MOVB    #VISEOL,(R0)+           ;MOVE 'EOL' OVER '1'
2378  005556  005304                          DEC     R4                      ;DONE '16' LINES
2379  005560  001366                          BNE     SHFEOL                  ;NO, SHIFT NEXT 'EOL'
2380  005562  104004                          DELAY                           ;3 SEC. DISPLAY DELAY
2381  005564  005305                          DEC     R5                      ;DONE '63' SHIFTS
2382  005566  001357                          BNE     SRHEOL                  ;NO, LOOP AGAIN
2383  005570  104006                          ENDTST
2384  005572  000730                          BR      VTEOL                   ;YES, RESTART TEST
2385                                  ;*************************************************************************
2386                                  ;VT 'END OF SCREEN' TEST (5)
2387                                  ;THIS TEST TESTS THAT THE 'VISIBLE END OF SCEEEN' CHARACTER CAN
2388                                  ;BE PLACED IN ALL '1024' CHARACTER POSITIONS ON THE SCREEN. THIS
2389                                  ;IS DONE BY FIRST PRE-LOADING THE  BUFFER TO BE DISPLAYED WITH
2390                                  ;QUESTION MARKS. THE 'EOS' CHARACTER IS THEN INCREMENTED THRJ THIS
2391                                  ;BUFFER REPLACING THE QUESTION MARKS BEHIND THE 'EOS' CHARACTER WITH
```

```
2392                                    ;DOTS.
2393                                    ;*************************************************************************
2394
2395  005574  012737  005574  020406  VTEOS:  MOV     #VTEOS, RETURN
2396  005602  104000                          DISPLAY
2397  005604  023341                          MES30                   ;TEXT 'VT END OF SCREEN TEST'
2398  005606  104011                          PRELOAD                 ;PRE-LOAD THE BUFFER WITH '?'
2399  005610     077                          .BYTE   77
2400  005611     077                          .BYTE   77
2401  005612  012701  002000                  MOV     #1024.,R1       ;SET UP CHAR. CNTR.
2402  005616  012700  024474                  MOV     #VTBUFF,R0      ;SET UP BUFFER POINTER
2403  005622  112710  000231                  MOVB    #VISEOS,(R0)    ;LOAD 1ST CHAR. WITH 'EOS'
2404  005626  012777  000001  173654          MOV     #1,@VTCSR       ;ST DISPLAY
2405  005634  104004                  SRHEOS: DELAY
2406  005636  005301                          DEC     R1              ;DONE
2407  005640  001405                          BEQ     ENDEOS          ;YES, EXIT
2408  005642  012720  000056                  MOV     #56,(R0)+       ;NO, REPLACE THE 'EOS' W/ DOT.
2409  005646  112710  000231                  MOVB    #VISEOS,(R0)    ;MOVE 'EOS' OVER '1' PLACE
2410  005652  000770                          BR      SRHEOS
2411  005654  104006                  ENDEOS: ENDTST
2412  005656  000746                          BR      VTEOS           ;YES, RESTART TEST
2413
2414                                    ;*************************************************************************
2415                                    ;VT 'BLANK CONTROL' TEST (6)
2416                                    ;THIS TEST INCREMENTS THE 'BLANK' CONTROL CHARACTER (361) THRU A
2417                                    ;FULL SCREEN BUFFER OF ?'S. THE SCREEN STARTS BLANK AND END'S
2418                                    ;UP FULL OF *'S.
2419                                    ;*************************************************************************
2420
2421  005660  012737  005660  020406  VTBLANK:MOV     #VTBLANK,RETURN
2422  005666  104000                          DISPLAY
2423  005670  023371                          MES31                   ;TEXT 'VT BLANKING' TEST
2424  005672  104011                          PRELOAD                 ;PRELOAD DATA BUFFER
2425  005674     077                          .BYTE   77              ;W/ ?'S
2426  005675     077                          .BYTE   77
2427  005676  112710  000361                  MOVB    #BLANK,(R0)     ;SET UP BLANK CONTROL CHAR.
2428  005702  012701  002000                  MOV     #1024.,R1       ;SET UP CNTR
2429  005706  012777  000001  173574          MOV     #1,@VTCSR       ;ST. DISPLAY
2430  005714  104004                          DELAY                   ;3 SEC. DISPLAY DELAY
2431  005716  112720  000052          MOVBLK: MOVB    #52,(R0)+       ;REPLACE, 'BLANK' CHAR. W/ *
2432  005722  112710  000361                  MOVB    #BLANK,(R0)     ;MOVE 'BLANK' CHAR. UP '1'
2433  005726  104004                          DELAY
2434  005730  005301                          DEC     R1              ;MOVED 'BLANK' THUR BUFFER
2435  005732  001371                          BNE     MOVBLK          ;NO
2436  005734  104006                          ENDTST
2437  005736  000750                          BR      VTBLANK         ;RESTART TEST
2438
2439                                    ;*************************************************************************
2440                                    ;VT 'ALIGNMENT' TEST (7)
2441                                    ;THIS TEST DISPLAYS A FULL SCREEN OF THE CHAR. 'E' TO ENABLE A VISJAL
2442                                    ;ALIGNMENT OF THE 'VT'.
2443                                    ;*************************************************************************
2444
2445  005740  012737  005740  020406  VTALIN: MOV     #VTALIN,RETURN
2446  005746  104000                          DISPLAY
2447  005750  023421                          MES33                   ;TEXT 'VT' 'ALIGNMENT' TEST
```

```
                                        PRELOAD                 ;PRELOAD 'VT' DATA BUFFER
                                        .BYTE   105             ;W/ E'S
                                        .BYTE   105
                                        INC     @VTCSR          ;ST. DISPLAY
                                        THEND
                                        BR      VTALIN
                                ;*******************************************************************************
                                ;VT 'FOCUS' TEST (10)
                                ;THIS TEST DISPLAYS A FULL SCREEN OF ALTERNATE *'S & 1'S TO ENABLE A VISUAL
                                ;FOCUS OF THE 'VT'.
                                ;*******************************************************************************

                                VTFOCUS:MOV     #VTFOCUS,RETURN
                                        DISPLAY
                                        MES34                   ;TEXT 'VT FOCUS TEST'
                                        PRELOAD                 ;LOAD 'VT' DATA BUFFER
                                        .BYTE   52              ;WITH *'S
                                        .BYTE   61              ;AND 1'S
                                        INC     @VTCSR          ;ST. DISPLAY
                                        THEND
                                        BR      VTFOCUS

                                ;*******************************************************************************
                                ;VT 'WORST CASE' TEST (11)
                                ;THIS TEST DISPLAYS A FULL SCREEN OF THE WORST  CASE PATTERN '125 & 252'.
                                ;*******************************************************************************

                                VTWORST:MOV     #VTWORST,RETURN
                                         DISPLAY
                                        MES35                   ;TEXT 'VT WORST  CASE TEST'
                                        PRELOAD                 ;LOAD 'VT' BUFFER W/ WORST CASE
                                        .BYTE   125             ;CHAR. 125
                                        .BYTE   252             ;CHAR. 252
                                        INC     @VTCSR          ;ST. DISPLAY
                                        THEND
                                        BR      VTWORST
                                ;*******************************************************************************
                                ;VT 'CURSOR MOVEMENT' TEST (12)
                                ;THIS TEST TESTS THAT THE CURSOR CONTROL CHARACTER CAN BE PLACED
                                ;IN ANY POSITION ON THE SCREEN.THIS IS DONE BY DISPLAYING A 'CURSOR EOS'
                                ;BEING INCREMENTED ACROSS THE SCREEN. AS THE CURSOR IS INCREMENTED ALONG
                                ;ASTRICKS ARE FILLED IN BEHIND IT AND WHEN THE TEST IS COMPLETE THE
                                ;ENTIRE SCREEN WILL BE FILLED WITH *'S.
                                ;*******************************************************************************

                                VTCURSR:MOV     #VTCURSR,RETURN
                                        DISPLAY
                                        MES36                   ;TEXT 'VT CURSOR MOVEMENT TEST'
                                        PRELOAD                 ;PRE-LOAD THE BUFFER WITH '?'
                                        .BYTE   77
                                        .BYTE   77
                                        MOV     #1024.,R1       ;SET UP CHAR. CNTR.
                                        MOV     #VTBUFF,R0      ;SET UP BUFFER POINTER
                                        MOV     R0,@VTCAR       ;LOAD 'CURSOR' ADDRESS REG.
                                        MOV     #VISEOS,(R0)    ;LOAD 1ST CHAR. WITH 'EOS'
```

```
2504  006100  012777  000001  173402          MOV     #1,@VTCSR       ;ST DISPLAY
2505  006106  104004                  SHFCUR: DELAY
2506  006110  005301                          DEC     R1              ;DONE
2507  006112  001407                          BEQ     ENDCUR          ;YES, EXIT
2508  006114  012720  000056                  MOV     #56,(R0)+       ;NO, REPLACE THE 'EOS' W/ DOT.
2509  006120  010077  173360                  MOV     R0,@VTCAR       ;UPDATE 'CAR' TO STAY WITH 'EOS'
2510  006124  012710  000231                  MOV     #VISEOS,(R0)    ;MOVE 'EOS' OVER '1' PLACE
2511  006130  000766                          BR      SHFCUR
2512  006132  104006                  ENDCUR: ENDTST
2513  006134  000742                          BR      VTCURSR         ;YES, RESTART TEST
2514
2515                                  ;*************************************************************************
2516                                  ;VT 'ODD ADDRESS' TEST (13)
2517                                  ;THIS TEST SIMPLY DISPLAYS A MESSAGE WHICH IS LOADED FROM AN ODD
2518                                  ;STARTING ADDRESS.
2519                                  ;*************************************************************************
2520
2521  006136  012737  006136  020406  VTODD:  MOV     #VTODD, RETURN
2522  006144  104000                          DISPLAY
2523  006146  022077                          MES10                   ;TEXT 'VT ODD ADDRESS TEST'
2524  006150  005077  173334                  CLR     @VTCSR          ;STOP DISPLAY
2525  006154  012777  022251  173324          MOV     #MES11,@VTSAR   ;LOAD 'SAR' W/ MESSAGE FROM ODD ADDRESS.
2526  006162  005277  173322                  INC     @VTCSR          ;START & DISPLAY MESSAGE
2527  006166  104023                          THEND
2528  006170  000762                          BR      VTODD
2529  006172  000421                          BR      FKLOGIC         ;CONTINUE TO FK LOGIC TEST
2530                                  ;*************************************************************************
2531                                  ;FK11 FUNCTION KEYBOARD LOGIC TEST (15)
2532                                  ;THIS IS AN OPERATOR INTERVENTION TEST WHICH TESTS THE FUNCTION KEYBOARD LOGIC.
2533                                  ;REQUESTS ARE MADE FROM THE PROGRAM FOR INPUTS FROM THE FUNCTION KEYBOARD.
2534                                  ;*************************************************************************
2535
2536  006174  013704  001474  SETUPF: MOV     FKCSR,FKCSR4            ;MOVE 'CSR' ADDR. TO R4
2537  006200  005037  024444          CLR     MESPRT                  ;CLR PRINT INHIBIT SW.
2538  006204  012737  000400  020402  MOV     #400,ICOUNT             ;INITIALIZE THE ITERATION COUNT
2539  006212  012737  017446  000034  MOV     #FKERR,@#34             ;LOAD 'TRAP' VECTOR FOR ERROR HANDLER
2540  006220  012737  006246  020406  MOV     #FKT1,RETURN            ;SET UP THE RETURN ADDRESS FOR SCOPE
2541  006226  012737  000001  024142  MOV     #1,TSTNUM               ;INIT TEST NO. CNTR.
2542  006234  000207                  RTS     PC
2543
2544  006236  104000          FKLOGIC:DISPLAY
2545  006240  021240                  MES2                            ;TEXT 'FK LOGIC TEST:
2546  006242  004737  006174          JSR     PC,SETUPF               ;INITIALIZE THE 'FK' LOGIC TEST
2547
2548                                  ;*************************************************************************
2549                                  ;ISSUE 'RESET' AND CHECK THAT THE 'CONTROL REG. IS INITIALIZED CORRECTLY
2550                                  ;*************************************************************************
2551
2552  006246  000240          FKT1:   NOP                             ;FK TEST '1'
2553  006250  000240                  NOP
2554  006252  000005                  RESET
2555  006254  005714                  TST     @FKCSR4                 ;'RESET' SHOULD OF CLEARED REGISTER
2556  006256  001401                  BEQ     FKT2                    ;BRANCH IF CLEARED
2557  006260  104400                  ERROR                           ;FK 'CSR' WASN'T CLEARED VIA ' RESET'
2558
2559                                  ;*************************************************************************
```

```
2560
2561
2562
2563  006262  104001  000002          ;TEST THAT THE 'ERROR', 'ASCII FLAG', & 'GO' BITS CAN BE WROTE TO '1'
2564  006266  052714  177665          ;*********************************************************************
2565  006272  011400
2566  006274  042700  002000  FKT2:    SCOPE,2                    ;'FK' TEST 2
2567  006300  022700  100201          BIS     #177665,@FKCSR4            ;WRITE ALL 'CSR' BITS EXCEPT '6' TO 1'S
2568  006304  001401          MOV     @FKCSR4,R0
2569  006306  104400          BIC     #2000,R0                           ;CLR INTERVAL TIMING BIT
2570  006310  012714  000012  CMP     #100201,R0                 ;TEST IF BITS '15,7,0' SET
2571                          BEQ     .+4
2572                          ERROR                              ;BITS '15,7&0' FAILED TO WRITE TO 1 IN 'CSR'
2573                          MOV     #12,@FKCSR4                ;CLR FLAGS
2574
2575                    ;*********************************************************************
2576  006314  104001  000003  ;TEST THAT THE  'GO' BIT CAN BE WROTE TO '0'
2577  006320  052714  177665  ;*********************************************************************
2578  006324  005014
2579  006326  032714  000001  FKT3:    SCOPE,3                    ;'FK' TEST 3
2580  006332  001401          BIS     #177665,@FKCSR4            ;WRITE ALL 'CSR' BITS TO '1'
2581  006334  104400          CLR     @FKCSR4                    ;WRITE BITS TO '0'
2582  006336  012714  000012  BIT     #1,@FKCSR4                 ;TEST IF 'GO' BIT CLEARED
2583                          BEQ     .+4
2584                          ERROR                              ;THE 'GO' BIT DIDN'T CLR
2585                          MOV     #12,@FKCSR4                ;CLR FLAGS
2586                    ;*********************************************************************
2587  006342  104001  000004  ;TEST THAT THE 'ERROR', 'ASCII FLAG' & 'GO' BITS CAN BE CLEARED VIA 'RESET'
2588  006346  052714  177665  ;*********************************************************************
2589  006352  000005
2590  006354  005714  FKT4:    SCOPE,4                    ;'FK' TEST 4
2591  006356  001401          BIS     #177665,@FKCSR4            ;WRITE BITS TO 1
2592  006360  104400          RESET
2593                          TST     @FKCSR4                    ;TEST IF 'RESET' CLEARED 'CSR'
2594                          BEQ     .+4
2595                          ERROR                              ;RESET FAILED TO CLR 'CSR'
2596
2597                    ;*********************************************************************
2598  006362  104001  000005  ;TEST THAT THE 'INTERVAL TIMER' FLAG DOESN'T SET IF 'GO' IS CLEARED
2599  006366  012700  176000  ;*********************************************************************
2600  006372  005200
2601  006374  001376  FKT5:    SCOPE,5                    ;'FK' TEST 5
2602  006376  032714  002000          MOV     #-2000,R0                  ;SET UP WAIT LOOP
2603  006402  001401          INC     R0                         ;WAIT FOR INTERVAL TIMER FLAG
2604  006404  104400          BNE     .-2
2605  006406  012714  000012  BIT     #2000,@FKCSR4
2606                          BEQ     .+4
2607                          ERROR                              ;INTERVAL TIMER FLAG SET W/GO CLEARED.
2608                          MOV     #12,@FKCSR4                ;CLR FLAGS
2609
2610                    ;*********************************************************************
2611  006412  104001  000006  ;TEST THAT THE 'INTERVAL TIMER' FLAG WILL SET IF GO IS SET
2612  006416  012700  176000  ;*********************************************************************
2613  006422  052714  000001
2614  006426  005200  FKT6:    SCOPE,6                    ;'FK' TEST 6
2615  006430  001376          MOV     #-2000,R0                  ;SET UP WAIT LOOP
                               BIS     #1,@FKCSR4                 ;SET 'GO'
                               INC     R0                         ;GIVE FLAG A CHANCE TO SET
                               BNE     .-2
```

```
2616  006432  032714  002000          BIT     #2000,@FKCSR4           ;TEST IF FLAG SET
2617  006436  001001                  BNE     .+4
2618  006440  104400                  ERROR                           ;INTERVAL TIMER FLAG DIDN'T SET W/GO SET
2619  006442  012714  000012          MOV     #12,@FKCSR4             ;CLR FLAGS
2620
2621                          ;*************************************************************************
2622                          ;TEST THAT THE INTERVAL TIMER FLAG CAN BE CLEARED VIA SETTING BIT 1
2623                          ;*************************************************************************
2624
2625  006446  104001  000007  FKT7:   SCOPE,7                  ;'FK' TEST 7
2626  006452  052714  000001          BIS     #1,@FKCSR4              ;SET 'GO'
2627  006456  032714  002000          BIT     #2000,@FKCSR4           ;WAIT FOR FLAG
2628  006462  001775                  BEQ     .-4
2629  006464  042714  000001          BIC     #1,@FKCSR4              ;CLR 'GO'
2630  006470  052714  000002          BIS     #2,@FKCSR4              ;CLR INTERVAL TIMER FLAG
2631  006474  032714  002000          BIT     #2000,@FKCSR4           ;SEE IF FLAG CLEARED
2632  006500  001401                  BEQ     .+4
2633  006502  104400                  ERROR                           ;SETTING BIT '1' DIDN'T CLR INTERVAL FLAG.
2634                          ;*************************************************************************
2635                          ;TEST THAT 'RESET' WILL CLR THE 'INTERVAL TIMER' FLAG.
2636                          ;*************************************************************************
2637
2638  006504  104001  000010  FKT10:  SCOPE,10                 ;'FK' TEST 10
2639  006510  052714  000001          BIS     #1,@FKCSR4              ;SET 'GO'
2640  006514  032714  002000          BIT     #2000,@FKCSR4           ;WAIT FOR FLAG
2641  006520  001775                  BEQ     .-4
2642  006522  000005                  RESET                           ;CLR FLAG W/RESET
2643  006524  005714                  TST     @FKCSR4                 ;RESET SHOULD OF CLEARED REG.
2644  006526  001401                  BEQ     .+4
2645  006530  104400                  ERROR                           ;'RESET' DIDN'T CLR INTERVAL FLAG
2646
2647                          ;*************************************************************************
2648                          ;TEST THAT THE INTERVAL TIMER FLAG CAUSES AN INTERRUPT WITH PROCESSOR PRIORITY @ 0
2649                          ;*************************************************************************
2650
2651  006532  104001  000011  FKT11:  SCOPE,11                 ;'FK' TEST 11
2652  006536  012700  176000          MOV     #-2000,R0               ;SET UP A WAIT LOOP
2653  006542  004737  016654          JSR     PC,LDFKVT               ;SET UP 'FK' INTR. ADDR.
2654  006546  006560                  TAGA                            ;INTERRUPT SERVICE ADDR.
2655  006550  005200                  INC     R0                      ;WAIT FOR INTERRUPT
2656  006552  001376                  BNE     .-2
2657  006554  104400                  ERROR                           ;INTERVAL TIMER FLAG DIDN'T CAUSE INTERRUPT
2658  006556  000401                  BR      .+4
2659  006560  022626          TAGA:   POP2SP                          ;RESET THE STACK
2660  006562  004737  016710          JSR     PC,CLRFKV               ;CLR 'FK' INTR. ADDR.
2661
2662                          ;*************************************************************************
2663                          ;TEST THAT 'NO' INTERRUPT OCCURRS IF 'GO' IS CLEARED
2664                          ;*************************************************************************
2665
2666  006566  104001  000012  FKT12:  SCOPE,12                 ;'FK' TEST 12
2667  006572  012777  006630  172700  MOV     #TAGAB,@FKINT           ;SET UP THE INTERRUPT ADDRESS
2668  006600  012777  000340  172674  MOV     #340,@FKLVL
2669  006606  012700  176000          MOV     #-2000,R0               ;SET UP A WAIT LOOP
2670  006612  005077  172474          CLR     @PSW                    ;CLR PROC. PRIORITY
2671  006616  012714  000100          MOV     #100,@FKCSR4            ;SET INTR. ENABLE
```

```
2672  006622  005200                    INC     R0
2673  006624  001376                    BNE     .-2
2674  006626  000402                    BR      .+6                     ;OK, NO INTERRUPT OCCURRED.
2675  006630  022626              TAGAB: POP2SP                          ;RESET THE STACK
2676  006632  104400                    ERROR                           ;INTERRUPT OCCURRED W/ 'GO' CLR
2677  006634  004737  016710            JSR     PC,CLRFKV               ;CLR 'FK' INTR. ADDR.
2678                              ;*************************************************************************
2679                              ;TEST THAT 'NO' INTERRUPT OCCURS WITH PROC. @ PRIORITY '4'
2680                              ;*************************************************************************
2681
2682  006640  104001  000013      FKT13:  SCOPE,13                        ;'FK' TEST 13
2683  006644  012777  006704  172626    MOV     #TAGB,@FKINT            ;SET UP THE INTERRUPT ADDRESS
2684  006652  012777  000340  172622    MOV     #340,@FKLVL
2685  006660  012700  176000            MOV     #-2000,R0               ;SET UP A WAIT LOOP
2686  006664  012777  000200  172420    MOV     #200,@PSW               ;SET PROC. @ PRIORITY '4'
2687  006672  012714  000101            MOV     #101,@FKCSR4            ;SET INTR. ENABLE & GO.
2688  006676  005200                    INC     R0
2689  006700  001376                    BNE     .-2
2690  006702  000402                    BR      .+6                     ;OK, NO INTERRUPT OCCURRED.
2691  006704  022626              TAGB:  POP2SP                          ;RESET THE STACK
2692  006706  104400                    ERROR                           ;INTERRUPT OCCURRED W/ PROC. @ PRIORITY '4'
2693  006710  004737  016710            JSR     PC,CLRFKV               ;CLR 'FK' INTR. ADDR.
2694                              ;*************************************************************************
2695                              ;TEST THAT 'NO' INTERRUPT OCCURRES W/ PROC. @ PRIORITY '5'
2696                              ;*************************************************************************
2697
2698  006714  104001  000014      FKT14:  SCOPE,14                        ;'FK' TEST 14
2699  006720  012777  006760  172552    MOV     #TAGC,@FKINT            ;SET UP THE INTERRUPT VECTOR
2700  006726  012777  000340  172546    MOV     #340,@FKLVL
2701  006734  012700  176000            MOV     #-2000,R0               ;SET UP WAIT LOOP
2702  006740  012777  000240  172344    MOV     #240,@PSW               ;SET PROC. @ PRIORITY '5'
2703  006746  012714  000101            MOV     #101,@FKCSR4            ;SET INTER. ENABLE & GO
2704  006752  005200                    INC     R0
2705  006754  001376                    BNE     .-2
2706  006756  000402                    BR      .+6                     ;OK, NO INTERRUPT OCCURRED
2707  006760  022626              TAGC:  POP2SP                          ;RESET STACK POINTER
2708  006762  104400                    ERROR                           ;FK INTERRUPTED WITH PROC. @ PRIORITY '5'
2709  006764  004737  016710            JSR     PC,CLRFKV               ;CLR 'FK' INTR. ADDR.
2710
2711                              ;*************************************************************************
2712                              ;TEST THAT THE INTERVAL TIMER FLAG IS SET APPROXIMATELY EVERY '500' USEC.
2713                              ;*************************************************************************
2714
2715  006770  104001  000015      FKT15:  SCOPE,15                        ;'FK' TEST 15
2716  006774  005001                    CLR     R1
2717  006776  004737  016654            JSR     PC,LDFKVT               ;SET UP THE INTR. ADDR.
2718  007002  007010                    TAGFA
2719  007004  005201                    INC     R1
2720  007006  001376                    BNE     .-2
2721
2722  007010  022626              TAGFA: POP2SP                          ;RESET STACK
2723  007012  004737  016710            JSR     PC,CLRFKV               ;CLR 'FK' VECTOR ADDR.
2724  007016  022701  000400            CMP     #400,R1                 ;CHECK LOW LIMIT
2725  007022  101402                    BLOS    .+6                     ;BRANCH IF HIGHER
2726  007024  104400                    ERROR                           ;INTERVAL CLOCK FREQ. TO FAST
2727  007026  000404                    BR      FKT16                   ;EXIT ON ERROR
```

```
2728  007030  022701  000600            CMP     #600,R1                 ;CHECK HIGH LIMIT
2729  007034  103001                    BHIS    .+4                     ;BRANCH IF LOWER
2730  007036  104400                    ERROR                           ;INTERVAL CLOCK FREQ. TO SLOW
2731                                ;*************************************************************************
2732                                ;TEST THAT 'CLR ASCII' (BIT3) WILL CLEAR THE KEYBOARD FLAG & ERROR BIT.
2733                                ;*************************************************************************
2734
2735  007040  104001  000016        FKT16:  SCOPE,16                        ;'FK' TEST 16
2736  007044  052714  100200                BIS     #100200,@FKCSR4         ;SET THE KEYBOARD FLAG
2737  007050  105714                        TSTB    @FKCSR4                 ;TEST THAT DONE WAS SET
2738  007052  100401                        BMI     .+4
2739  007054  104400                        ERROR                           ;'DONE' DIDN'T SET
2740  007056  052714  000010                BIS     #10,@FKCSR4             ;SET 'CLR FLAG'
2741  007062  032714  100200                BIT     #100200,@FKCSR4         ;TEST IF FLAGS CLEARED
2742  007066  001401                        BEQ     .+4
2743  007070  104400                        ERROR                           ;SETTING BIT 3 DIDN'T CLR ASCII FLAGS
2744
2745                                ;*************************************************************************
2746                                ;TEST THAT THE KEYBOARD FLAG IS CLEARED WHEN THE KEYBOARD DATA IS READ
2747                                ;*************************************************************************
2748
2749  007072  104001  000017        FKT17:  SCOPE,17                        ;'FK' TEST 17
2750  007076  052714  000201                BIS     #201,@FKCSR4            ;SET THE KEYBOARD FLAG
2751  007102  005777  172370                TST     @FKDATA                 ;READ KEYBOARD DATA
2752  007106  105714                        TSTB    @FKCSR4                 ;RE-TEST IF FLAG CLEARED
2753  007110  100001                        BPL     .+4
2754  007112  104400                        ERROR                           ;READING KEYBOARD DATA DIDN'T CLR FLAG
2755  007114  012714  000012                MOV     #12,@FKCSR4             ;CLR FLAGS
2756
2757                                ;*************************************************************************
2758                                ;TEST THAT THE 'ERROR' FLAG IS CLEARED WHEN THE KEYBOARD DATA IS READ.
2759                                ;*************************************************************************
2760
2761  007120  104001  000020        FKT20:  SCOPE,20                        ;'FK' TEST 20
2762  007124  052714  100001                BIS     #100001,@FKCSR4         ;SET THE ERROR BIT
2763  007130  005714                        TST     @FKCSR4                 ;TEST THAT ERROR SET
2764  007132  100401                        BMI     .+4
2765  007134  104400                        ERROR                           ;ERROR WASN'T SET
2766  007136  005777  172334                TST     @FKDATA                 ;READ THE KEYBOARD DATA
2767  007142  005714                        TST     @FKCSR4                 ;RE-TEST THE ERROR FLAG
2768  007144  100001                        BPL     .+4
2769  007146  104400                        ERROR                           ;READING KEYBOARD DATA DIDN'T CLR 'ERROR'
2770  007150  005014                        CLR     @FKCSR4                 ;CLR 'GO'
2771  007152  052714  000012                BIS     #12,@FKCSR4             ;CLR FLAGS
2772  007156  104001                        SCOPE
2773  007160  000020                        20
2774  007162  005037  020402                CLR     ICOUNT                  ;RUN FOLLOWING TESTS '1' PASS ONLY
2775                                ;*************************************************************************
2776                                ;TEST THAT THE KEYBOARD FLAG CAN BE SET VIA THE KEYBOARD.
2777                                ;*************************************************************************
2778
2779  007166  104001  000021        FKT21:  SCOPE,21                        ;'FK' TEST 21
2780  007172  032777  002000  172124        BIT     #2000,@SWR              ;IS MANUAL INHIBIT SW. SET?
2781  007200  001076                        BNE     FKT24+4                 ;YES, TEST COMPLETE
2782  007202  012737  007210  020406        MOV     #TAGFH,RETURN           ;RE-SET SCOPE ADDRESS
2783  007210  052714  000001        TAGFH:  BIS     #1,@FKCSR4              ;SET 'GO'
```

```
2784  007214  104000                  DISPLAY
2785  007216  021506                  MES6                            ;REQUEST CHAR.
2786  007220  005077  172264          CLR     @VTCSR                  ;CLEAR MESSAGE
2787  007224  012700  177773          MOV     #-5,R0
2788  007230  005001          TAGH:   CLR     R1
2789  007232  005201                  INC     R1
2790  007234  001376                  BNE     .-2
2791  007236  005200                  INC     R0
2792  007240  001373                  BNE     TAGH
2793  007242  105714                  TSTB    @FKCSR4                 ;TEST IF KEYBOARD FLAG SET
2794  007244  100401                  BMI     .+4
2795  007246  104400                  ERROR                           ;KEYBOARD FLAG DIDN'T SET
2796  007250  012714  000012          MOV     #12,@FKCSR4             ;CLR FLAGS
2797
2798                          ;*****************************************************************
2799                          ;TEST THAT THE 'ASCII' FLAG ISN'T SET IF 'GO' IS CLEARED
2800                          ;*****************************************************************
2801
2802  007254  104001  000022  FKT22:  SCOPE,22                        ;'FK' TEST 22
2803  007260  104000                  DISPLAY
2804  007262  021534                  MES6A                           ;REQUEST CHAR.
2805  007264  005077  172220          CLR     @VTCSR                  ;CLEAR MESSAGE
2806  007270  012700  177773          MOV     #-5,R0
2807  007274  005001          TAGAH:  CLR     R1
2808  007276  005201                  INC     R1
2809  007300  001376                  BNE     .-2
2810  007302  005200                  INC     R0
2811  007304  001373                  BNE     TAGAH
2812  007306  105714                  TSTB    @FKCSR4                 ;TEST IF KEYBOARD FLAG SET
2813  007310  100001                  BPL     .+4
2814  007312  104400                  ERROR                           ;ASCII FLAG SET W/ 'GO'CLEARED
2815  007314  012714  000012          MOV     #12,@FKCSR4             ;CLR FLAGS
2816                          ;*****************************************************************
2817                          ;TEST THAT THE 'ERROR' FLAG IS SET IF THE KEYBOARD FLAG ISN'T CLEARED
2818                          ;*****************************************************************
2819
2820  007320  104001  000023  FKT23:  SCOPE,23                        ;'FK' TEST 23
2821  007324  012714  000201          MOV     #201,@FKCSR4            ; SET 'GO'
2822  007330  104000                  DISPLAY
2823  007332  021562                  MES6B                           ;REQUEST CHAR.
2824  007334  005077  172150          CLR     @VTCSR                  ;CLEAR MESSAGE
2825  007340  012700  177773          MOV     #-5,R0
2826  007344  005001          TAGI:   CLR     R1
2827  007346  005201                  INC     R1
2828  007350  001376                  BNE     .-2
2829  007352  005200                  INC     R0
2830  007354  001373                  BNE     TAGI
2831  007356  005714                  TST     @FKCSR4                 ;TEST IF 'ERROR SET'
2832  007360  100401                  BMI     .+4
2833  007362  104400                  ERROR                           ;ERROR DIDN'T SET ON 2ND CHAR.
2834  007364  005014                  CLR     @FKCSR4                 ;CLR 'GO'
2835  007366  052714  000012          BIS     #12,@FKCSR4             ;CLR FLAGS
2836
2837                          ;*****************************************************************
2838                          ;END OF 'FK' LOGIC TEST.
2839                          ;*****************************************************************
```

```
2840
2841  007372  104001  000024          FKT24:  SCOPE,24                        ;'FK' TEST 24
2842  007376  104000                          DISPLAY
2843  007400  021610                          MES7                            ;TEXT 'LOGIC OK:
2844  007402  104006                          ENDTST                          ;END OF TEST
2845  007404  000402                          BR      .+6
2846  007406  000137  003122                  JMP     VTLOGIC                 ;RESTART THE 'VT' LOGIC TEST
2847  007412  000137  006236                  JMP     FKLOGIC                 ;RE-RUN 'FK' LOGIC TEST
2848                                  ;*************************************************************************
2849                                  ;FK11 FUNCTION KEYBOARD LIGHT & SWITCH TEST. (16)
2850                                  ;THIS IS AN OPERATOR INTERVENTION TEST WHICH MONITORS THE FUNCTION KEYS.
2851                                  ;WHEN A FUNCTION KEY OR KEYS ARE STRUCK ITS CORRESPONDING FUNCTION LIGHT
2852                                  ;IS LIT.  IF THE KEY IS STRUCK A SECOND TIME THE CORRESPONDING FUNCTION
2853                                  ;LIGHT IS TURNED OFF.
2854                                  ;*************************************************************************
2855
2856  007416  012737  007416  020406  FKFUN:  MOV     #FKFUN, RETURN
2857  007424  104000                          DISPLAY
2858  007426  021105                          MES1                            ;TEXT 'FK FUNCTION KEYBOARD LIGHT TEST'
2859  007430  005000                          CLR     R0                      ;CLR WORKINGS REG'S
2860  007432  005001                          CLR     R1
2861  007434  005002                          CLR     R2
2862  007436  004737  016654                  JSR     PC,LDFKVT               ;LOAD THE INTR. ADDR.
2863  007442  007450                          SRRVFK                          ;INTR SERVICE ROUTINE
2864  007444  000001                          WAIT                            ;WAIT FOR INTERRUPT
2865  007446  000776                          BR      .-2
2866
2867                                  ;ROUTINE TO 'SERVICE' FUNCTION KEYBOARD INTERRUPTS
2868
2869  007450  005202                  SRRVFK: INC     R2                      ;INTR. CNTR. (EVERY 1000, LITE DATA UPDATED)
2870  007452  105777  172016                  TSTB    @FKCSR
2871  007456  100004                          BPL     1$
2872  007460  022777  000003  172010          CMP     #3,@FKDATA              ;CONTROL C TYPED?
2873  007466  001436                          BEQ     EXITF1                  ;IF SO-EXITF1
2874  007470  022702  001000          1$:     CMP     #1000,R2                        ;SERVICED '512' FLAGS?
2875  007474  001023                          BNE     EXITF0                  ;NO, EXIT
2876  007476  005002                          CLR     R2                      ;CLR CNTR.
2877  007500  017704  171766                  MOV     @FKLDA,R4               ;READ SWITCH DATA 'A'
2878  007504  001406                          BEQ     SRVFOB                  ;SW. DATA PRESENT?
2879  007506  010037  024436                  MOV     R0,SAVOLD               ;SAVE 'OLD' KEYBOARD DATA
2880  007512  040400                          BIC     R4,R0                   ;CLR ALL CORRESPONDING BITS IN 'OLD' NO.
2881  007514  043704  024436                  BIC     SAVOLD,R4               ;CLR ALL CORRESPONDING BITS IN 'NEW' NO.
2882  007520  050400                          BIS     R4,R0                   ;SET REMAINING BITS FOR 'NEW' VALUE
2883  007522  017704  171742          SRVFOB: MOV     @FKLDB,R4               ;READ SWITCH DATA 'B'
2884  007526  001406                          BEQ     EXITF0                  ;SW. DATA PRESENT?
2885  007530  010137  024436                  MOV     R1,SAVOLD
2886  007534  040401                          BIC     R4,R1                   ;CLR ALL CORRESPONDING BITS IN 'OLD' WRD.
2887  007536  043704  024436                  BIC     SAVOLD,R4               ;CLR ALL CORRESPONDING BITS IN 'NEW WRD.
2888  007542  050401                          BIS     R4,R1                   ;UPDATE 'OLD' WRD WITH ' NEW' WRD.
2889  007544  010077  171722          EXITF0: MOV     R0,@FKLDA               ;TRANSMIT UPDATED LIGHT DATA 'A'
2890  007550  010177  171714                  MOV     R1,@FKLDB               ;TRANSMIT UPDATED LIGHT DATA 'B'
2891  007554  052777  000012  171712          BIS     #12,@FKCSR              ;CLR FLAGS.
2892  007562  000002                          RTI
2893  007564  052777  000012  171702  EXITF1: BIS     #12,@FKCSR
2894  007572  104006                          ENDTST
2895  007574  000710                          BR      FKFUN
```

```
2896                            ;*****************************************************************************
2897                            ;KEYBOARD CHARACTER TEST (17)
2898                            ;THIS TEST VERIFIES THAT ALL CHARACTERS CAN BE TRANSMITTED VIA THE FK11 KEYBOARD.
2899                            ;WHEN THE TEST IS STARTED, THE TEXT "READY FOR INPUT" IS DISPLAYED ON THE 'VT'.
2900                            ;THE OPERATOR THEN TYPES IN THREE PASSES OF THE ENTIRE KEYBOARD
2901                            ;(LOWER CASE, SHIFT AND CONTROL) STARTING AT THE TOP LEFT HAND SIDE OF THE
2902                            ;KEYBOARD FOR EACH PASS.  EACH CHARACTER TRANSMITTED FROM THE KEYBOARD IS VERIFIED
2903                            ;AGAINST A CHARACTER BUFFER AND DISPLAYED ON THE 'VT'.  IF THE TRANSMITTED
2904                            ;CHARACTER DOESN'T MATCH THE EXPECTED CHARACTER AN AUDIO 'BEEP' IS HEARD
2905                            ;AND THE CHARACTER IS DISPLAYED IN BOLD UNDERLINED.  ALL CHARACTERS RECEIVED
2906                            ;FROM THIS POINT ON WILL BE TRANSMITTED IN THIS MODE UNTIL EITHER THE CORRECT
2907                            ;CHARACTER CODE IS RECEIVED OR DATA 'SW15' IS SET TO A '1' ALLOWING THE PROGRAM
2908                            ;TO CONTINUE.  DATA 'SW14' MAY ALSO BE SET TO A '1' AT ANYTIME TO LOOP ON
2909                            ;THE CURRENT CHARACTER.
2910                            ;*****************************************************************************
2911
2912  007576  012737  007576  020406  FKCHAR: MOV     #FKCHAR,RETURN
2913  007604  104000                    DISPLAY
2914  007606  021312                    MES4                        ;TEXT 'FK CHARACTER TEST'
2915  007610  012701  020463            MOV     #CHRTAB-1,R1        ;SET UP CHAR. TABLE POINTER
2916  007614  005037  024144            CLR     FIELDSW             ;CLR FIELD CONTROL SOFTWARE SW.
2917  007620  005037  024440            CLR     TEMP1               ;SET UP SOFTWARE SW.'S
2918  007624  005037  024442            CLR     TEMP2
2919  007630  005037  024460            CLR     KSTOR1
2920  007634  004737  016654            JSR     PC,LDFKVT           ;LOAD 'FK' VECTOR ADDR.
2921  007640  007646                    CHKFKO
2922  007642  000001                    WAIT                        ;WAIT FOR 'FK' INTERRUPTS
2923  007644  000776                    BR      .-2
2924
2925                            ;ENTERED HERE ON 'FK' INTERRUPTS
2926
2927  007646  105777  171622  CHKFKO: TSTB    @FKCSR              ;ASCII FLAG SET?
2928  007652  100071                    BPL     EXTCHR              ;NO, INTERVAL TIMER FLAG
2929  007654  117702  171616            MOVB    @FKDATA,R2          ;READ & SAVE DATA
2930  007660  032777  040000  171436    BIT     #SW14,@SWR          ;IS THE LOOP SW. SET?
2931  007666  001007                    BNE     CHKFK1+2            ;YES, DON'T UPDATE POINTER
2932  007670  005737  024144            TST     FIELDSW             ;NO, IS THE ERROR FLAG SET?
2933  007674  001403                    BEQ     CHKFK1              ;NO, UPDATE POINTER
2934  007676  005777  171422            TST     @SWR                ;YES, IS SCOPE SW. SET?
2935  007702  100001                    BPL     CHKFK1+2            ;YES, LOOP ON CURRENT CHAR.
2936  007704  105721          CHKFK1: TSTB    (R1)+               ;NO, UPDATE POINTER
2937  007706  120211                    CMPB    R2,(R1)             ;DOES RECV'D CHAR.=EXPT'D CHAR.
2938  007710  001073                    BNE     CHRERR              ;NO CHAR. ERROR
2939  007712  005737  024144            TST     FIELDSW             ;IS FIELD SW. SET?
2940  007716  001404                    BEQ     CHKFK2              ;NO
2941  007720  005037  024144            CLR     FIELDSW             ;YES, CLR IT
2942  007724  112720  000360            MOVB    #360,(R0)+          ;CLR FIELD CONTROL CHAR.
2943  007730  022702  000033  CHKFK2: CMP     #33,R2
2944  007734  001403                    BEQ     .+10
2945  007736  005737  024460            TST     KSTOR1              ;DOING CONTROL PASS?
2946  007742  001402                    BEQ     .+6                 ;NO, CONTINUE
2947  007744  052702  000100            BIS     #100,R2             ;YES, ADD BIT SO CHAR. IS VISIBLE
2948  007750  110220                    MOVB    R2,(R0)+            ;MOV RECV'D CHAR. TO 'VT' BUFFER.
2949  007752  005737  024440            TST     TEMP1
2950  007756  001012                    BNE     CHKFK4
2951  007760  022701  020551            CMP     #UPCASE-1,R1        ;STARTING 'UPPER CASE' PASS?
```

```
2952  007764  001007                          BNE     CHKFK4          ;NO
2953  007766  104000                          DISPLAY
2954  007770  023077                          MES22                   ;TEXT "INPUT UPPER CASE"
2955  007772  012737  007576  024130          MOV     #FKCHAR,AVECTR  ;RE-SET '↑A' RESTART ADDR.
2956  010000  005237  024440                  INC     TEMP1           ;SET PRINT INHIBIT SW.
2957  010004  005737  024442  CHKFK4: TST     TEMP2
2958  010010  001012                          BNE     EXTCHR
2959  010012  022701  020637                  CMP     #CONTRL-1,R1    ;STARTING CONTROL PASS?
2960  010016  001007                          BNE     EXTCHR          ;NO
2961  010020  104000                          DISPLAY
2962  010022  023143                          MES23                   ;TEXT "INPUT CONROL CHAR.S"
2963  010024  012737  007576  024130          MOV     #FKCHAR,AVECTR  ;RE-SET '↑A' RESTART ADDR.
2964  010032  005237  024442                  INC     TEMP2           ;SET DISPLAY INHIBIT SW.
2965  010036  022701  020653  EXTCHR: CMP     #CNTRIT,R1      ;STARTING CONTROL PASS
2966  010042  001002                          BNE     .+6             ;NO, CONTINUE
2967  010044  005237  024460                  INC     KSTOR1          ;YES, SET UP TO DISPLAY INVISIBLE CHAR.
2968  010050  052777  000002  171416          BIS     #2,@FKCSR       ;CLR INTERVAL TIMER FLAG
2969  010056  022701  020726                  CMP     #CHREND+1,R1    ;FINISHED?
2970  010062  001401                          BEQ     .+4             ;YES, RE-START TEST
2971  010064  000002                          RTI                     ;EXIT
2972  010066  004737  016710                  JSR     PC,CLRFKV       ;CLR INTERRUPT VECTOR
2973  010072  022626                          POP2SP                  ;RESET STACK
2974  010074  104006                          ENDTST
2975  010076  000637                          BR      FKCHAR          ;RESTART TEST
2976  010100  052777  000200  171402  CHRERR: BIS     #200,@VTCSR     ;ISSUE AUDIO BEEP ON ERROR
2977  010106  005737  024144                  TST     FIELDSW         ;IS FIELD SW SET?
2978  010112  001306                          BNE     CHKFK2          ;YES
2979  010114  005237  024144                  INC     FIELDSW         ;NO, SET 'FIELD' SW.
2980  010120  112720  000374                  MOVB    #374,(R0)+      ;SELECT 'BOLD-UNDERLINE'
2981  010124  000701                          BR      CHKFK2          ;SET FIELD CONTROL SOFTWARE SW.
2982                        ;*****************************************************************
2983                        ;FUNCTION KEYBOARD REPEATIBILITY TEST  20)
2984                        ;THIS TEST VERIFIES THAT A SELECTED CHARACTER OR CHARACTERS (MAXIMUM OF TWO)
2985                        ;ARE RECEIVED CORRECTLY FROM THE KEYBOARD.  WHEN THE 'REPEATIBILITY' TEST IS
2986                        ;SELECTED THE PROGRAM HALTS.  LOAD THE CODE FOR THE CHARACTER TO BE VERIFIED
2987                        ;IN THE DATA SWITCHES AND PRESS CONTINUE.  THE PROGRAM WILL HALT AGAIN.
2988                        ;LOAD THE CODE FOR THE 2ND CHARACTER TO BE VERIFIED.  IF ONLY
2989                        ;ONE CHARACTER IS BEING CHECKED, PRESS CONTINUE THE 2ND TIME WITHOUT CHANG-
2990                        ;ING THE SWITCH SETTING.  AS INPUTTED CHARACTERS ARE VERIFIED THEY ARE
2991                        ;ALSO DISPLAYED ON THE 'VT'.  IF AN ERROR IS DETECTED, THE AUDIO 'BEEP' IS
2992                        ;SOUNDED AND THE RECEIVED CHARACTER IS DISPLAYED AS 'BOLD-UNDERLINED'.
2993                        ;*****************************************************************
2994
2995  010126  005737  024154  FKREPT: TST     TTYSWH          ;TTY AVAILABLE?
2996  010132  001404                          BEQ     FKRPTA          ;NO, HALT TO GET CHAR. CODE
2997  010134  104012                          PRINT                   ;YES,
2998  010136  022506                          MES15                   ;REQUEST CHAR. CODES
2999  010140  104015                          ASEMBL                  ;DECODE CODES
3000  010142  000406                          BR      FKRPTB
3001  010144  000000          FKRPTA: HALT                    ;GET 1ST CHAR. FROM THE SWITCHES
3002  010146  117703  171152                  MOVB    @SWR,R3         ;SAVE 1ST CHAR.
3003  010152  000000                          HALT                    ;GET 2ND CHAR. FROM THE SWITCHES
3004  010154  117704  171144                  MOVB    @SWR,R4         ;SAVE '2ND' CHAR.
3005  010160  042703  000200  FKRPTB: BIC     #200,R3
3006  010164  042704  000200                  BIC     #200,R4
3007  010170  012737  010160  020406          MOV     #FKRPTB,RETURN
```

```
3008  010176  104000                        DISPLAY
3009  010200  022433                        MES14                   ;TEXT 'FK REPEATIBILITY TEST'
3010  010202  012737  010126  024130        MOV     #FKREPT,AVECTR
3011  010210  005037  024144                CLR     FIELDSW         ;CLR FIELD CONTROL SOFTWARE SW.
3012  010214  005005                        CLR     R5              ;CLR WORKING REG.
3013  010216  004737  017062                JSR     PC,TTYENB       ;ENABLE TTY INTERRUPTS
3014  010222  004737  016654                JSR     PC,LDFKVT       ;LOAD 'FK' INTR. ADDR.
3015  010226  010234                        FREPTO
3016  010230  000001                        WAIT                    ;WAIT FOR 'FK' INTR.
3017  010232  000776                        BR      .-2
3018                                ;ENTERED HERE ON 'FK' INTERRUPTS FOR THE REPEATIBILITY TEST
3019
3020  010234  105777  171234        FREPTO: TSTB    @FKCSR          ;ASCII FLAG SET?
302:  010240  100035                        BPL     EXREPT          ;NO, INTERVAL TIMER FLAG
3022  010242  117701  171230                MOVB    @FKDATA,R1      ;YES, READ & SAVE DATA
3023  010246  005705                        TST     R5              ;COMPARE 1ST OR 2ND CHAR.?
3024  010250  001003                        BNE     FREPT1          ;2ND
3025  010252  120103                        CMPB    R1,R3           ;RECV'D = EXPT'D CHAR.?
3026  010254  001416                        BEQ     REPTOK          ;YES
3027  010256  000402                        BR      REPTER          ;NO, SET UP ERROR
3028  010260  120104        FREPT1: CMPB    R1,R4           ;RECV'D = EXPT'D CHAR.?
3029  010262  001413                        BEQ     REPTOK          ;YES
3030  010264  052777  000200  171216 REPTER: BIS     #AUDIO,@VTCSR   ;ISSUE AUDIO BEEP
3031  010272  005737  024144                TST     FIELDSW         ;FIELD SW. SET?
3032  010276  001014                        BNE     FREPT2          ;YES
3033  010300  005237  024144                INC     FIELDSW         ;NO, SET IT
3034  010304  112720  000374                MOVB    #374,(R0)+      ;SEND BAD CHAR. AS 'BOLD-UNDERLINED'
3035  010310  000407                        BR      FREPT2
3036  010312  005737  024144        REPTOK: TST     FIELDSW         ;FIELD SW. SET
3037  010316  001404                        BEQ     FREPT2          ;YES
3038  010320  005037  024144                CLR     FIELDSW         ;NO, CLR IT
3039  010324  112720  000360                MOVB    #360,(R0)+      ;CLR FIELD CONTROL CHAR.
3040  010330  005105                FREPT2: COM     R5              ;SET UP SOFTWARE SW.
3041  010332  110120                        MOVB    R1,(R0)+        ;TRANSMIT RECV'D CHAR.
3042  010334  052777  000002  171132 EXREPT: BIS     #2,@FKCSR       ;CLR INTERVAL TIMER FLAG
3043  010342  000002                        RTI
3044                                ;*****************************************************************************
3045                                ;VT20 SYSTEM EXERCISER TEST. (TEST 21)
3046                                ;THIS TEST IS DESIGNED TO EXERCISE THE VT20 IN A SYSTEM ENVIRONMENT. THE
3047                                ;TEST UTILIZES THE VT20 (DISPLAYS & KEYBOARDS) AND ALL ASSOCIATED DL11'S
3048                                ;IN CONJUNCTION WITH A HOST COMPUTER. TO RUN THIS TEST THE DL11 DATA
3049                                ;HANDLING ROUTINE MUST BE LOADED INTO THE HOST COMPUTER. THIS ROUTINE
3050                                ;RECEIVES AND RE-TRANSMITS DATA TO VT20 BEING TESTED. WHEN THE SYSTEM TEST
3051                                ;IS STARTED THE PROGRAM WAITS FOR CHARACTERS TO BE INPUTTED FROM THE
3052                                ;KEYBOARD.  AS THESE CHARACTERS ARE RECEIVED THEY ARE STORED IN A VT BUFFER
3053                                ;AND DISPLAYED. THE PROGRAM WILL ACCEPT UP TO 384 CHARACTERS (6 COMPLETE
3054                                ;LINES) FROM THE KEYBOARD. THESE 384 CHARACTERS, OR ANY PART THERE OF,
3055                                ;CAN BE AUTOMATICALLY GENERATED BY TYPING A '↑A'. THE OPERATOR MAY AT
3056                                ;THIS TIME OR ANYTIME PRIOR TO THIS, TYPE 'ALTMODE'. THIS WILL ENABLE
3057                                ;THE INPUTTED CHARACTERS TO BE TRANSMITTED TO THE HOST COMPUTER.
3058                                ;UPON RECEIVING THE COMPLETE DATA TRANSFER, THE HOST COMPUTER WILL
3059                                ;RE-TRANSMIT THE DATA BACK TO THE VT20. THE RECEIVED DATA IS THEN
3060                                ;VERIFIED AGAINST THE DATA ORGINALLY TRANSMITTED AND THEN DISPLAYED.
3061                                ;IF THE RECEIVED DATA DOESN'T MATCH THE TRANSMITTED DATA, THE RECEIVED
3062                                ;CHACACTER IS DISPLAYED AS 'BOLD-UNDERLINED'.
3063                                ;THE SYSTEM TEST USES AN ERROR REPORTING SCHEME OF DISPLAYING ALL
```

```
3064                                    ;ERRORS AS THEY ARE ENCOUNTERED AS UNIQUE CURSOR CHARACTERS. THESE
3065                                    ;CHARACTERS IDENTIFY THE PARTICULAR ERROR DETECTED. TYPING A '↑C' AT
3066                                    ;ANYTIME WILL ENABLE THE TEST TO BE RE-STARTED.
3067                                    ;TYPING A '↑T' AFTER THE DATA HAS BEEN INPUTTED WILL ENABLE
3068                                    ;FOR A CONTINUOUS TRANSFER OF DATA.  TYPING '↑T' A SECOND TIME
3069                                    ;WILL STOP THE DATA TRANSFER (ON COMPLETION OF CURRENT TRANSFER)
3070                                    ;AND ALLOW THE PROGRAM TO RESUME NORMAL OPERATION.
3071                                    ;*****************************************************************************
3072
3073  010344  012737  010344  020406  SYSTST: MOV     #SYSTST,RETURN
3074  010352  000005                          RESET
3075  010354  012737  010344  024130          MOV     #SYSTST,AVECTR          ;SET UP THE RESTART ADDR.
3076  010362  012777  021651  171022          MOV     #MES9,@VTOSAR           ;DISPLAY TEST HEADER ON UNIT '0'
3077  010370  012777  021651  171030          MOV     #MES9,@VT1SAR           ;DISPLAY TEST HEADER ON UNIT '1'
3078  010376  022737  000004  024136          CMP     #4,     UNITNO  ;4 UNITS?
3079  010404  001013                          BNE     2$
3080  010406  012777  021651  171042          MOV     #MES9,  @VT3SAR ;DISPLAY TEST HEADER ON UNIT '3'
3081  010414  005277  171040                  INC     @VT3CSR
3082  010420  012777  021651  171014  1$:     MOV     #MES9,  @VT2SAR ;DISPLAY TEST HEADER ON UNIT '2'
3083  010426  005277  171012                  INC     @VT2CSR
3084  010432  000404                          BR      3$
3085
3086  010434  022737  000003  024136  2$:     CMP     #3,     UNITNO  ;3 UNITS?
3087  010442  001766                          BEQ     1$
3088  010444  005277  170744          3$:     INC     @VTOCSR
3089  010450  005277  170754                  INC     @VT1CSR
3090  010454  104020                          DELAYL
3091  010456  104020                          DELAYL
3092  010460  104005                          EOSBUF                          ;RE-LOAD DATA BUFFER W/ 'EOS'
3093  010462  012777  000340  170622  SYSIT1: MOV     #340,@PSW               ;SET PROC. PRIO. @7
3094  010470  012700  024170                  MOV     #RECSWO,R0              ;SET UP TO CLR SOFTWARE USER SW.'S
3095  010474  005020                          CLR     (R0)+
3096  010476  022700  024424                  CMP     #START,R0               ;DONE?
3097  010502  001374                          BNE     .-6                     ;NO
3098  010504  012777  011766  171026          MOV     #RECV0,@RINT0           ;LOAD DL11 REC. VECTOR ADDR.'S
3099  010512  012777  000200  171022          MOV     #200,@RLVL0             ;BR LEVEL '4'
3100  010520  012777  012000  171016          MOV     #RECV1,@RINT1
3101  010526  012777  000200  171012          MOV     #200,@RLVL1
3102  010534  012777  011706  171036          MOV     #TRAN0,@XINT0           ;LOAD DL11 TRANS. VECTOR ADDR.'S
3103  010542  012777  000200  171032          MOV     #200,@XLVL0
3104  010550  012777  011722  171026          MOV     #TRAN1,@XINT1
3105  010556  012777  000200  171022          MOV     #200,@XLVL1
3106  010564  012777  011626  170546          MOV     #SERFK0,@FK0INT         ;LOAD 'FK' VECTOR ADDRESSES
3107  010572  012777  000200  170542          MOV     #200,@FK0LVL
3108  010600  012777  011642  170546          MOV     #SERFK1,@FK1INT
3109  010606  012777  000200  170542          MOV     #200,@FK1LVL
3110  010614  012700  024474                  MOV     #VTBUFF,R0              ;LOAD 'VT' BUFFER POINTERS
3111  010620  010002                          MOV     R0,R2
3112  010622  004737  011610                  JSR     PC,DISPMS               ;DISPLAY HEADER MESSAGE
3113  010626  012702  026444                  MOV     #VTBUFF+1000.,R2
3114  010632  010077  170554                  MOV     R0,@VTOSAR              ;LOAD 'VT' STARTING ADDR. REG.'S
3115  010636  010277  170564                  MOV     R2,@VT1SAR
3116  010642  004737  011610                  JSR     PC,DISPMS
3117  010646  012700  024563                  MOV     #VTBUFF+55.,R0
3118  010652  010037  024404                  MOV     R0,BUF0R0               ;R0 BUFFER POINTER FOR UNIT '0'
3119  010656  010037  024414                  MOV     R0,BUF0R1               ;R1 BUFFER POINTER FOR UNIT '0'
```

```
3120  010662  012702  026533                 MOV     #VTBUFF+1055.,R2
3121  010666  010237  024406                 MOV     R2,BUF1R0               ;R0 BUFFER POINTER FOR UNIT '1'
3122  010672  010237  024416                 MOV     R2,BUF1R1               ;R1 BUFFER POINTER FOR UNIT '1'
3123  010676  012737  010726  000004         MOV     #DL11A,@#4      ;SET UP TIME-OUT FOR MISSING DL'S
3124  010704  012737  000340  000006         MOV     #340,@#6
3125  010712  012777  000100  170600         MOV     #100,@RCSR0             ;ENABLE DL11 REC. INTR'S
3126  010720  012777  000100  170632         MOV     #100,@XCSR0             ;ENABLE DL11 TRANS, INTR.'S
3127  010726  012737  010750  000004  DL11A: MOV     #DL11B,@#4
3128  010734  012777  000100  170560         MOV     #100,@RCSR1
3129  010742  012777  000100  170612         MOV     #100,@XCSR1
3130  010750  022737  000004  024136  DL11B: CMP     #4,UNITNO               ;4 UNITS?
3131  010756  001133                         BNE     TAG2X                   ;NO, TEST FOR 3
3132  010760  012777  012024  170566         MOV     #RECV3, @RINT3          ;LOAD DL11 REC VECTOR FOR UNIT '3'
3133  010766  012777  000200  170562         MOV     #200,@RLVL3             ;BR LEVEL '4'
3134  010774  012777  011752  170612         MOV     #TRAN3,@XINT3           ;LOAD DL11 XMIT VECTOR FOR UNIT '3'
3135  011002  012777  000200  170606         MOV     #200,@XLVL3             ;BR LEVEL '4'
3136  011010  012777  011672  170366         MOV     #SERFK3,@FK3INT         ;LOAD FK VECTOR FOR UNIT '3'
3137  011016  012777  000200  170362         MOV     #200,@FK3LVL            ;BR LEVEL '4'
3138  011024  012702  032364                 MOV     #VTBUFF+3000.,R2
3139  011030  010277  170422                 MOV     R2,@VT3SAR              ;LOAD 'VT'3 STARTING ADDR. REG
3140  011034  004737  011610                 JSR     PC,DISPMS
3141  011040  012702  032453                 MOV     #VTBUFF+3055.,R2
3142  011044  010237  024412                 MOV     R2,     BUF3R0          ;BUFFER POINTER FOR UNIT '3'
3143  011050  010237  024422                 MOV     R2,     BUF3R1
3144  011054  012737  011076  000004         MOV     #DL11C,@#4
3145  011062  012777  000100  170436         MOV     #100,@RCSR3             ;ENABLE DL11 FOR UNIT '3' TO INTERRUPT
3146  011070  012777  000100  170470         MOV     #100,   @XCSR3
3147  011076  012777  000101  170354  DL11C: MOV     #101,@VT3CSR            ;START VT
3148  011104  012777  000101  170266         MOV     #101,@FK3CSR            ;ENABLE FK INTERRUPT
3149  011112  012777  012012  170430  TAG1X: MOV     #RECV2,@RINT2           ;LOAD DL11 REC VECTOR FOR UNIT '2'
3150  011120  012777  000200  170424         MOV     #200,@RLVL2             ;BR LEVEL '4'
3151  011126  012777  011736  170454         MOV     #TRAN2,@XINT2           ;LOAD DL11 XMIT VECTOR FOR UNIT '2'
3152  011134  012777  000200  170450         MOV     #200,@XLVL2             ;BR LEVEL '4'
3153  011142  012777  011656  170220         MOV     #SERFK2,@FK2INT         ;LOAD FK VECTOR FOR UNIT '2'
3154  011150  012777  000200  170214         MOV     #200,@FK2LVL            ;BR LEVEL '4'
3155  011156  012702  030414                 MOV     #VTBUFF+2000.,R2
3156  011162  010277  170254                 MOV     R2,@VT2SAR              ;LOAD 'VT'2 STARTING ADDR. REG
3157  011166  004737  011610                 JSR     PC,DISPMS
3158  011172  012702  030503                 MOV     #VTBUFF+2055.,R2
3159  011176  010237  024410                 MOV     R2,BUF2R0               ;BUFFER POINTER FOR UNIT '2'
3160  011202  010237  024420                 MOV     R2,BUF2R1
3161  011206  012737  011230  000004         MOV     #DL11D,@#4
3162  011214  012777  000100  170302         MOV     #100,   @RCSR2  ;ENABLE DL11 FOR UNIT '2' TO INTERRUPT
3163  011222  012777  000100  170334         MOV     #100,@XCSR2
3164  011230  012777  000101  170206  DL11D: MOV     #101,@VT2CSR            ;START VT
3165  011236  012777  000101  170120         MOV     #101,@FK2CSR            ;ENABLE FK INTERRUPTS
3166  011244  000404                         BR      TAG3X
3167
3168  011246  022737  000003  024136  TAG2X: CMP     #3,UNITNO               ;3 UNITS?
3169  011254  001716                         BEQ     TAG1X                   ;YES, SET UP UNIT 2
3170  011256  012737  000006  000004  TAG3X: MOV     #6,@#4                  ;RESET THE TIMEOUT VECTOR
3171  011264  012737  000004  000006         MOV     #4,@#6
3172  011272  012777  000101  170114         MOV     #101,@VT0CSR            ;START VT'S
3173  011300  012777  000101  170122         MOV     #101,@VT1CSR
3174  011306  012777  000101  170020         MOV     #101,@FK0CSR            ;ENABLE 'FK' INTERRUPTS
3175  011314  012777  000101  170026         MOV     #101,@FK1CSR
```

```
3176  011322  005077  167764                  CLR     @PSW                    ;SET PROC. PRIC. @C
3177
3178  011326  012700  024504          SERVCE: MOV     #VTBUFF+08.,R0
3179  011332  013702  024230                  MOV     CHRCT0,R2
3180  011336  104024                          BINDEC                          ;UPDATE THE CHARACTER COUNTER
3181  011340  012700  024522                  MOV     #VTBUFF+22.,R0
3182  011344  013702  024314                  MOV     RECTR0,R2
3183  011350  104024                          BINDEC                          ;UPDTAE THE RECV'D CHAR. COUNTER
3184  011352  012700  024540                  MOV     #VTBUFF+36.,R0
3185  011356  013702  024304                  MOV     TRANF0,R2
3186  011362  104024                          BINDEC                          ;JPDATE THE TRANSFER COUNTER
3187  011364  012700  024555                  MOV     #VTBUFF+49.,R0
3188  011370  013702  024334                  MOV     ERFLG0,R2
3189  011374  104024                          BINDEC                          ;UPDATE THE ERROR COUNTER
3190  011376  012700  026454                  MOV     #VTBUFF+1008.,R0
3191  011402  013702  024232                  MOV     CHRCT1,R2
3192  011406  104024                          BINDEC
3193  011410  012700  026472                  MOV     #VTBUFF+1022.,R0
3194  011414  013702  024316                  MOV     RECTR1,R2
3195  011420  104024                          BINDEC
3196  011422  012700  026510                  MOV     #VTBUFF+1036.,R0
3197  011426  013702  024306                  MOV     TRANF1,R2
3198  011432  104024                          BINDEC
3199  011434  012700  026525                  MOV     #VTBUFF+1049.,R0
3200  011440  013702  024336                  MOV     ERFLG1,R2
3201  011444  104024                          BINDEC
3202  011446  023727  024136  000003          CMP     UNITNO,#3               ;RUNNING '3' UNITS ?
3203  011454  002724                          BLT     SERVCE                  ;NO, RETURN TO SERVICE
3204  011456  012700  030424                  MOV     #VTBUFF+2008.,R0
3205  011462  013702  024234                  MOV     CHRCT2,R2
3206  011466  104024                          BINDEC                          ;UPDATE THE CHARACTER COUNTER
3207  011470  012700  030442                  MOV     #VTBUFF+2022.,R0
3208  011474  013702  024320                  MOV     RECTR2,R2
3209  011500  104024                          BINDEC                          ;UPDTAE THE RECV'D CHAR. COUNTER
3210  011502  012700  030460                  MOV     #VTBUFF+2036.,R0
3211  011506  013702  024310                  MOV     TRANF2,R2
3212  011512  104024                          BINDEC                          ;UPDATE THE TRANSFER COUNTER
3213  011514  012700  030475                  MOV     #VTBUFF+2049.,R0
3214  011520  013702  024340                  MOV     ERFLG2,R2
3215  011524  104024                          BINDEC                          ;UPDATE THE ERROR COUNTER
3216  011526  023727  024136  000004          CMP     UNITNO,#4               ;RUNNING '4' UNITS ?
3217  011534  002674                          BLT     SERVCE                  ;NO, RETURN TO SERVICE
3218  011536  012700  032374                  MOV     #VTBUFF+3008.,R0
3219  011542  013702  024236                  MOV     CHRCT3,R2
3220  011546  104024                          BINDEC
3221  011550  012700  032412                  MOV     #VTBUFF+3022.,R0
3222  011554  013702  024322                  MOV     RECTR3,R2
3223  011560  104024                          BINDEC
3224  011562  012700  032430                  MOV     #VTBUFF+3036.,R0
3225  011566  013702  024312                  MOV     TRANF3,R2
3226  011572  104024                          BINDEC
3227  011574  012700  032445                  MOV     #VTBUFF+3049.,R0
3228  011600  013702  024342                  MOV     ERFLG3,R2
3229  011604  104024                          BINDEC
3230  011606  000647                          BR      SERVCE
3231
```

```
3232                                    ;ENTERED HERE TO MOVE A MESSAGE INTO THE VT BUFFER
3233
3234  011610  012701  022157          DISPMS: MOV     #MES80,R1
3235  011614  111122                          MOVB    (R1),(R2)+
3236  011616  122721  000031                  CMPB    #EOS,(R1)+
3237  011622  001374                          BNE     DISPMS+4
3238  011624  000207                          RTS     PC
3239                                    ;ENTERED HERE ON 'FK' INTERRUPTS FROM UNIT '0'
3240
3241  011626  004737  012036          SERFK0: JSR     PC,SETBF0               ;SET UP BUFFER POINTERS
3242  011632  004737  012444                  JSR     PC,SERVFK               ;SERVICE 'FK' INTERRUPT
3243  011636  000137  012272                  JMP     SAVBF0                  ;SAVE BUFFER POINTERS ON RETURN
3244
3245                                    ;ENTERED HERE ON 'FK' INTERRUPTS FROM UNIT '1'
3246
3247  011642  004737  012102          SERFK1: JSR     PC,SETBF1               ;SET UP BUFFER POINTERS
3248  011646  004737  012444                  JSR     PC,SERVFK               ;SERVICE 'FK' INTERRUPT
3249  011652  000137  012316                  JMP     SAVBF1                  ;SAVE BUFFER POINTERS ON RETURN
3250
3251                                    ;ENTERED HERE ON 'FK' INTERRUPTS FROM UNIT '2'
3252
3253  011656  004737  012152          SERFK2: JSR     PC,SETBF2               ;SET UP BUFFER POINTERS
3254  011662  004737  012444                  JSR     PC,SERVFK               ;SERVICE 'FK' INTERRUPT
3255  011666  000137  012342                  JMP     SAVBF2                  ;SAVE BUFFER POINTERS
3256
3257                                    ;ENTERED HERE ON 'FK' INTERRUPTS FROM UNIT '3'
3258
3259  011672  004737  012222          SERFK3: JSR     PC,SETBF3               ;SET UP BUFFER POINTERS
3260  011676  004737  012444                  JSR     PC,SERVFK               ;SERVICE 'FK' INTERRUPT
3261  011702  000137  012366                  JMP     SAVBF3                  ;SAVE BUFFER POINTERS
3262                                    ;ENTERED HERE ON DL11 TRANSMITTER INTERRUPTS FROM UNIT '0'
3263
3264  011706  004737  012036          TRAN0:  JSR     PC,SETBF0               ;SET UP BUFFER POINTERS
3265  011712  004737  013130                  JSR     PC,TRNSMT
3266  011716  000137  012272                  JMP     SAVBF0
3267
3268                                    ;ENTERED HERE ON DL11 TRANSMITTER INTERRUPTS FROM UNIT '1'
3269
3270  011722  004737  012102          TRAN1:  JSR     PC,SETBF1               ;SET UP BUFFER POINTERS
3271  011726  004737  013130                  JSR     PC,TRNSMT
3272  011732  000137  012316                  JMP     SAVBF1
3273
3274                                    ;ENTERED HERE ON DL11 TRANSMITTER INTERRUPTS FROM UNIT '2'
3275
3276  011736  004737  012152          TRAN2:  JSR     PC,SETBF2               ;SET UP BUFFER POINTERS
3277  011742  004737  013130                  JSR     PC,TRNSMT               ;SERVICE TRANSMITTER INTERRUPT
3278  011746  000137  012342                  JMP     SAVBF2
3279
3280                                    ;ENTERED HERE ON DL11 TRANSMITTER INTERRUPTS FROM UNIT '3'
3281
3282  011752  004737  012222          TRAN3:  JSR     PC,SETBF3               ;SET UP BUFFER POINTERS
3283  011756  004737  013130                  JSR     PC,TRNSMT
3284  011762  000137  012366                  JMP     SAVBF3
3285
3286                                    ;ENTEPED HERE ON DL11 RECEIVER INTERRUPTS FROM UNIT '0'
3287
```

```
3288  011766  004737  012036          RECV0:  JSR     PC,SETBF0
3289  011772  004737  013362                  JSR     PC,RECVIT
3290  011776  000535                          BR      SAVBF0
3291                                  ;ENTERED HERE ON DL11 RECEIVER INTERRUPTS FROM UNIT '1'
3292
3293  012000  004737  012102          RECV1:  JSR     PC,SETBF1
3294  012004  004737  013362                  JSR     PC,RECVIT
3295  012010  000542                          BR      SAVBF1
3296
3297                                  ;ENTERED HERE ON DL11 RECEIVER INTERRUPTS FROM UNIT '2'
3298
3299  012012  004737  012152          RECV2:  JSR     PC,SETBF2               ;SET UP BUFFER POINTERS
3300  012016  004737  013362                  JSR     PC,RECVIT               ;SERVICE RECEIVER INTERRUPT
3301  012022  000547                          BR      SAVBF2
3302
3303                                  ;ENTERED HERE ON DL11 RECEIVER INTERRUPTS FROM UNIT '3'
3304
3305  012024  004737  012222          RECV3:  JSR     PC,SETBF3               ;SET UP BUFFER POINTERS
3306  012030  004737  013362                  JSR     PC,RECVIT               ;SERVICE RECEIVER INTERRUPT
3307  012034  000554                          BR      SAVBF3
3308
3309                                  ;SUBROUTINE ENTERED TO SET UP BUFFER POINTERS FOR UNIT '0'
3310
3311  012036  012637  024460          SETBF0: MOV     (SP)+,KSTOR1            ;SAVE THE RETURN ADDRESS
3312  012042  010046                          MOV     R0,-(SP)                ;SAVE THE WORKING REG.'S
3313  012044  010146                          MOV     R1,-(SP)
3314  012046  010246                          MOV     R2,-(SP)
3315  012050  010346                          MOV     R3,-(SP)
3316  012052  010446                          MOV     R4,-(SP)
3317  012054  013700  024404                  MOV     BUFOR0,R0
3318  012060  013701  024414                  MOV     BUFOR1,R1
3319  012064  005002                          CLR     R2                      ;R2 IS INDEX OFFSET FOR ADDRESSSING
3320  012066  005003                          CLR     R3                      ;R3 IS INDEX OFFSET FOR DEVICE ADDRESSING
3321  012070  012737  024563  024424          MOV     #VTBUFF+55.,START
3322  012076  000177  012356                  JMP     @KSTOR1                 ;RETURN TO CALL
3323
3324                                  ;SUBROUTINE ENTERED TO SET UP BUFFER POINTERS FOR UNIT '1'
3325
3326  012102  012637  024460          SETBF1: MOV     (SP)+,KSTOR1            ;SAVE THE RETURN ADDRESS
3327  012106  010046                          MOV     R0,-(SP)                ;SAVE THE WORKING REG.'S
3328  012110  010146                          MOV     R1,-(SP)
3329  012112  010246                          MOV     R2,-(SP)
3330  012114  010346                          MOV     R3,-(SP)
3331  012116  010446                          MOV     R4,-(SP)
3332  012120  013700  024406                  MOV     BUF1R0,R0
3333  012124  013701  024416                  MOV     BUF1R1,R1
3334  012130  012702  000002                  MOV     #2,R2                   ;R2 IS INDEX OFFSET FOR ADDRESSSING
3335  012134  012703  000014                  MOV     #12.,R3                 ;R3 IS INDEX OFFSET FOR DEVICE ADDRESSING
3336  012140  012737  026533  024424          MOV     #VTBUFF+1055.,START
3337  012146  000177  012306                  JMP     @KSTOR1                 ;RETURN TO CALL
3338
3339                                  ;SUBROUTINE ENTERED TO SET UP BUFFER POINTERS FOR UNIT '2'
3340
3341  012152  012637  024460          SETBF2: MOV     (SP)+,KSTOR1            ;SAVE THE RETURN ADDRESS
3342  012156  010046                          MOV     R0,-(SP)                ;SAVE THE WORKING REG.'S
3343  012160  010146                          MOV     R1,-(SP)
```

```
3344  012162  010246                        MOV     R2,-(SP)
3345  012164  010346                        MOV     R3,-(SP)
3346  012166  010446                        MOV     R4,-(SP)
3347  012170  013700  024410                MOV     BUF2R0,R0
3348  012174  013701  024420                MOV     BUF2R1,R1
3349  012200  012702  000004                MOV     #4,R2                   ;R2 IS INDEX OFFSET FOR ADDRESSING
3350  012204  012703  000030                MOV     #24.,R3                 ;R3 IS INDEX OFFSET FOR DEVICE ADDR.
3351  012210  012737  030503  024424        MOV     #VTBUFF+2055.,START
3352  012216  000177  012236                JMP     @KSTOR1                 ;RETURN TO CALL
3353
3354                                ;SUBROUTINE ENTERED TO SET UP BUFFER POINTERS FOR UNIT '3'
3355
3356  012222  012637  024460        SETBF3: MOV     (SP)+,KSTOR1            ;SAVE THE RETURN ADDRESS
3357  012226  010046                        MOV     R0,-(SP)                ;SAVE THE WORKING REG.'S
3358  012230  010146                        MOV     R1,-(SP)
3359  012232  010246                        MOV     R2,-(SP)
3360  012234  010346                        MOV     R3,-(SP)
3361  012236  010446                        MOV     R4,-(SP)
3362  012240  013700  024412                MOV     BUF3R0,R0
3363  012244  013701  024422                MOV     BUF3R1,R1
3364  012250  012702  000006                MOV     #6,R2
3365  012254  012703  000044                MOV     #36.,R3
3366  012260  012737  032453  024424        MOV     #VTBUFF+3055.,START
3367  012266  000177  012166                JMP     @KSTOR1                 ;RETURN TO CALL
3368
3369                                ;SUBROUTINE ENTERED TO SAVE BUFFER POINTERS FOR UNIT '0'
3370
3371  012272  010037  024404        SAVBF0: MOV     R0,BUF0R0
3372  012276  010137  024414                MOV     R1,BUF0R1
3373  012302  012604                        MOV     (SP)+,R4                ;RESTORE THE WORKING REG.'S
3374  012304  012603                        MOV     (SP)+,R3
3375  012306  012602                        MOV     (SP)+,R2
3376  012310  012601                        MOV     (SP)+,R1
3377  012312  012600                        MOV     (SP)+,R0
3378  012314  000002                        RTI
3379
3380                                ;SUBROUTINE ENTERED TO SAVE BUFFER POINTERS FOR UNIT '1'
3381
3382  012316  010037  024406        SAVBF1: MOV     R0,BUF1R0
3383  012322  010137  024416                MOV     R1,BUF1R1
3384  012326  012604                        MOV     (SP)+,R4                ;RESTORE THE WORKING REG.'S
3385  012330  012603                        MOV     (SP)+,R3
3386  012332  012602                        MOV     (SP)+,R2
3387  012334  012601                        MOV     (SP)+,R1
3388  012336  012600                        MOV     (SP)+,R0
3389  012340  000002                        RTI
3390
3391                                ;SUBROUTINE ENTERED TO SAVE BUFFER POINTERS FOR UNIT '2'
3392
3393  012342  010037  024410        SAVBF2: MOV     R0,BUF2R0
3394  012346  010137  024420                MOV     R1,BUF2R1
3395  012352  012604                        MOV     (SP)+,R4                ;RESTORE THE WORKING REG.'S
3396  012354  012603                        MOV     (SP)+,R3
3397  012356  012602                        MOV     (SP)+,R2
3398  012360  012601                        MOV     (SP)+,R1
3399  012362  012600                        MOV     (SP)+,R0
```

```
3400  012364  000002                    RTI
3401
3402                                    ;SUBROUTINE ENTERED TO SAVE BUFFER POINTERS FOR UNIT '3'
3403
3404  012366  010037  024412    SAVBF3: MOV     R0,BUF3R0
3405  012372  010137  024422            MOV     R1,BUF3R1
3406  012376  012604                    MOV     (SP)+,R4            ;RESTORE THE WORKING REG.'S
3407  012400  012603                    MOV     (SP)+,R3
3408  012402  012602                    MOV     (SP)+,R2
3409  012404  012601                    MOV     (SP)+,R1
3410  012406  012600                    MOV     (SP)+,R0
3411  012410  000002                    RTI
3412
3413                                    ;SUBROUTINE ENTERED ON ENCOUNTERING SYSTEM ERRORS WHICH MOVES
3414                                    ;THE CHARACTER IN THE PC INTO THE DISPLAY BUFFER.
3415
3416  012412  005262  024334    FLGERR: INC     ERFLG0(R2)          ;SET ERROR FLAG
3417  012416  112720  000362            MOVB    #BLINK,(R0)+        ;SET UP TO BLINK CHAR.
3418  012422  117620  000000            MOVB    @(SP),(R0)+         ;MOVE CHAR. INTO BUFFER
3419  012426  112720  000360            MOVB    #360,(R0)+          ;CLR PRESET FIELD
3420  012432  112710  000031            MOVB    #EOS,(R0)           ;TERMINATE W/ EOS.
3421  012436  062716  000002            ADD     #2,(SP)             ;SET UP STACK TO EXIT
3422  012442  000207                    RTS     PC
3423
3424                                    ;FK INTERRUPT SERVICE ROUTINE
3425
3426  012444  105773  001334    SERVFK: TSTB    @FKOCSR(R3)         ;ASCII FLAG SET?
3427  012450  100421                    BMI     SERVOO              ;YES, SERVICE KEYBOARD
3428  012452  005762  024274            TST     AUTSW0(R2)          ;IS THE AUTO CHAR. SW. SET?
3429  012456  001012                    BNE     AUTOLL              ;YES, GENERATE A CHARACTER
3430  012460  005762  024250            TST     PATERN(R2)          ;GENERATING WORST CASE PATTERN?
3431  012464  001011                    BNE     JMPWOR              ;YES, OUTPUT NEXT CHAR.
3432  012466  005762  024240            TST     HOLDS0(R2)          ;INTERVAL TIMER, IS ERROR HOLD SET?
3433  012472  001402                    BEQ     LINKOD              ;NO, CLR INTERVAL TIME FLAG
3434  012474  004737  013760            JSR     PC,TESTSW           ;YES, TST SWIS TO CONTINUE
3435  012500  000137  013106    LINKOD: JMP     SERVOD              ;SWIS STILL SET, EXIT
3436  012504  000137  013304    AUTOLL: JMP     AUTOUT
3437  012510  000137  013350    JMPWOR: JMP     WORST1
3438
3439  012514  005773  001334    SERVOO: TST     @FKOCSR(R3)         ;ERROR FLAG SET?
3440  012520  100003                    BPL     SERVOA              ;NO VALID CHAR.
3441  012522  004737  012412            JSR     PC,FLGERR           ;YES, ASCII ERRR FLAG SET
3442  012526  000113                    113                         ;DISPLAY A 'BLINKING K'
3443  012530  117304  001336    SERVOA: MOVB    @FKODAT(R3),R4      ;READ & SAVE CHAR.
3444  012534  122704  000003            CMPB    #3,R4               ;CHAR='↑C'?
3445  012540  001051                    BNE     SERVAA              ;NO, CONTINUE
3446  012542  005762  024374            TST     BUSY0(R2)           ;YES, DL'S BUSY?
3447  012546  001403                    BEQ     RELOAD              ;NO, RE-START
3448  012550  005262  024324            INC     STOP0(R2)           ;YES, SET STOP SW.
3449  012554  000554                    BR      SERVOD              ;EXIT
3450  012556  013700  024424    RELOAD: MOV     START,R0            ;RESET BUFFER POINTER
3451  012562  112710  000031            MOVB    #EOS,(R0)           ;SET UP FOR NEW INPUT
3452  012566  005062  024364            CLR     XFER0(R2)
3453  012572  005062  024354            CLR     RESTR0(R2)
3454  012576  005062  024324            CLR     STOP0(R2)
3455  012602  005062  024220            CLR     STCOD0(R2)
```

```
3456  012606  005062  024374                  CLR     BUSYO(R2)
3457  012612  005062  024262                  CLR     BLKERR(R2)
3458  012616  005062  024250                  CLR     PATERN(R2)
3459  012622  005062  024274                  CLR     AUTSWO(R2)
3460  012626  005062  024304                  CLR     TRANFO(R2)
3461  012632  005062  024314                  CLR     RECTRO(R2)
3462  012636  005062  024334                  CLR     ERFLGO(R2)
3463  012642  005062  024240                  CLR     HOLDSO(R2)
3464  012646  005062  024230                  CLR     CHRCTO(R2)              ;CLR CHAR. CNTR.
3465  012652  005062  024200                  CLR     TRNSWO(R2)
3466  012656  005062  024170                  CLR     RECSWO(R2)
3467  012662  000511                          BR      SERVOD                  ;EXIT
3468  012664  122704  000024          SERVAA: CMPB    #24,R4                  ;CHAR='↑T' ?
3469  012670  001017                          BNE     SERVOB                  ;NO, CONTINUE
3470  012672  005162  024364                  COM     XFERO(R2)               ;YES, COMPLIMENT TRANSFER SW.
3471  012676  001503                          BEQ     SERVOD                  ;STOP AUTO TRANSFER.
3472  012700  005762  024374                  TST     BUSYO(R2)               ;CURRENTLY TRANSFERRING DATA?
3473  012704  001100                          BNE     SERVOD                  ;YES, IGNORE SW.
3474  012706  005762  024354          SERVAB: TST     RESTRO(R2)              ;HAS A RESTART ADDR BEEN LOADED?
3475  012712  001403                          BEQ     .+10                    ;NO, TERMINATE INPUT BUFFER
3476  012714  016200  024354                  MOV     RESTRO(R2),R0           ;YES, RESET BUFFER POINTER
3477  012720  000423                          BR      SERVBB                  ;SET UP TO START TRANSFER
3478  012722  112720  000014                  MOVB    #EOP,(R0)+              ;TERMINATE MESSAGE
3479  012726  000420                          BR      SERVBB
3480
3481  012730  122704  000001          SERVOB: CMPB    #1,R4                   ;CHAR. = TO '↑A'
3482  012734  001556                          BEQ     AUTOST                  ;YES, AUTOMATICALLY GENERATE CHAR.'S
3483  012736  122704  000005                  CMPB    #5,R4                   ;CHAR. = TO '↑E'
3484  012742  001705                          BEQ     RELOAD                  ;YES, RELOAD
3485  012744  122704  000027                  CMPB    #27,R4                  ;CHAR. = TO '↑W' ?
3486  012750  001574                          BEQ     WORST                   ;YES, GENERATE WORST CASE PATTERN
3487  012752  005762  024374                  TST     BUSYO(R2)               ;DL11 BUSY?
3488  012756  001053                          BNE     SERVOD                  ;YES, IGNORE CHAR.
3489  012760  122704  000033                  CMPB    #33,R4                  ;REQUEST TRANSFER?
3490  012764  001024                          BNE     SERVOC                  ;NO
3491  012766  000747                          BR      SERVAB
3492  012770  112710  000031          SERVBB: MOVB    #EOS,(R0)
3493  012774  010062  024354                  MOV     R0,RESTRO(R2)           ;SAVE LAST ADDR. FOR AUTO RESTART
3494  013000  012762  000001  024374          MOV     #1,BUSYO(R2)            ;SET BUSY SW.
3495  013006  012762  000001  024200          MOV     #1,TRNSWO(R2)           ;SET TRANSFER, SOFTWARE SW.
3496  013014  013701  024424                  MOV     START,R1                ;SET UP BUFFER POINTER TO TRANSMIT DATA.
3497  013020  105741                          TSTB    -(R1)                   ;SET UP BUFFER POINTER
3498  013022  012762  000004  024210          MOV     #4,NULCTO(R2)
3499  013030  105072  001570          SERVBC: CLRB    @XBUFO(R2)              ;TRANSMIT THE START CODE
3500  013034  000424                          BR      SERVOD                  ;EXIT
3501  013036  122704  000177          SERVOC: CMPB    #177,R4                 ;CHAR. = 'RUBOUT'?
3502  013042  001010                          BNE     SERVCC                  ;NO, SAVE IT
3503  013044  005762  024230                  TST     CHRCTO(R2)              ;BUFFER EMPTY?
3504  013050  001416                          BEQ     SERVOD                  ;YES, IGNORE IT
3505  013052  005362  024230                  DEC     CHRCTO(R2)              ;NO, DECEMENT COUNTER
3506  013056  112740  000031                  MOVB    #EOS,-(R0)              ;BACK UP BUFFER POINTER
3507  013062  000411                          BR      SERVOD                  ;EXIT
3508  013064  026227  024230  000600  SERVCC: CMP     CHRCTO(R2),#384.        ;BUFFER FULL?
3509  013072  103011                          BHIS    SERVDX                  ;YES, IGNORE CHAR.
3510  013074  005262  024230                  INC     CHRCTO(R2)              ;INCREMENT CHAR. CNTR.
3511  013100  110420                          MOVB    R4,(R0)+                ;NO, SAVE CHAR.
```

```
3512  013102  112710  000031                  MOVB    #EOS,(R0)
3513  013106  052773  000002  001334  SERVOD: BIS     #2,@FKOCSR(R3)          ;CLR INTERVAL TIMER FLAG
3514  013114  000207                          RTS     PC                      ;EXIT
3515  013116  005062  024250          SERVDX: CLR     PATERN(R2)              ;CLEAR WORST CASE PATTERN SW.
3516  013122  005062  024274                  CLR     AUTSW0(R2)              ;CLEAR AUTO SW.
3517  013126  000767                          BR      SERVOD
3518
3519                                  ;DL11 TRANSMITTER SERVICE ROUTINE
3520
3521  013130  005762  024344          TRNSMT: TST     INTRN0(R2)              ;LEGAL INTERRUPT?
3522  013134  001003                          BNE     .+10                    ;YES, SERVICE IT.
3523  013136  005262  024344                  INC     INTRN0(R2)              ;THIS IS THE INITIALIZE INTERRUPT.
3524  013142  000207                          RTS     PC                      ;EXIT
3525  013144  005762  024200                  TST     TRNSW0(R2)              ;TRANSFER SW. SET?
3526  013150  001004                          BNE     TRANOA                  ;YES, LEGAL INTERRUPT
3527  013152  004737  012412                  JSR     PC,FLGERR               ;NO, ILLEGAL TRANSMITTER INTERRUPT
3528  013156  000124                          124                             ;DISPLAY A 'BLINKING T'.
3529  013160  000207                          RTS     PC
3530  013162  017737  166136  013270  TRANOA: MOV     @SWR,R3PRT              ;READ THE SWITCHES
3531  013170  042737  177774  013270          BIC     #177774,R3PRT
3532  013176  022737  000003  013270          CMP     #SW00+SW01,R3PRT        ;LOOPING DATA?
3533  013204  001430                          BEQ     TRANOC+4                ;YES, EXIT
3534  013206  005362  024210                  DEC     NULCT0(R2)
3535  013212  003306                          BGT     SERVBC                  ;XMITTED 4 NULL CHARS?
3536  013214  002406                          BLT     TRANOB
3537  013216  112772  000377  001570          MOVB    #377,@XBUF0(R2)
3538  013224  005062  024314                  CLR     RECTR0(R2)              ;RECV'D CHARACTER COUNTER
3539  013230  000207                          RTS PC
3540  013232  122721  000014          TRANOB: CMPB    #EOP,(R1)+              ;HAS LAST CHAR. BEEN TRANSMITTED?
3541  013236  001011                          BNE     TRANOC                  ;NO, TRANSMIT NEXT CHAR.
3542  013240  005062  024200                  CLR     TRNSW0(R2)              ;YES, CLR TRANSMIT SW.
3543  013244  005062  024262                  CLR     BLKERR(R2)              ;CLEAR BLOCK ERROR SW.
3544  013250  005262  024170                  INC     RECSW0(R2)              ;SET UP TO REC. DATA
3545  013254  013701  024424                  MOV     START,R1                ;RESET POINTER TO COMPARE DATA
3546  013260  000207                          RTS     PC                      ;EXIT
3547  013262  111172  001570          TRANOC: MOVB    (R1),@XBUF0(R2)         ;TRANSMIT NEXT CHAR.
3548  013266  000207                          RTS     PC
3549  013270  000000                  R3PRT:  .WORD   0
3550
3551                                  ;ENTERED HERE TO INITALIZE THE AUTOMATIC CHARACTER GENERATOR
3552
3553  013272  012704  000041          AUTOST: MOV     #41,R4
3554  013276  010462  024274                  MOV     R4,AUTSW0(R2)
3555  013302  000670                          BR      SERVCC
3556
3557                                  ;ENTERED HERE TO GENERATE AND OUTPUT THE AUTOMATIC CHARACTERS
3558
3559  013304  005262  024274          AUTOUT: INC     AUTSW0(R2)              ;UPDATE THE CHARACTER
3560  013310  042762  177600  024274          BIC     #177600,AUTSW0(R2)
3561  013316  032762  000140  024274          BIT     #140,AUTSW0(R2)         ;IS CHARACTER LEGAL?
3562  013324  001003                          BNE     .+10                    ;YES
3563  013326  052762  000040  024274          BIS     #40,AUTSW0(R2)          ;NO, RESET IT
3564  013334  016204  024274                  MOV     AUTSW0(R2),R4
3565  013340  000651                          BR      SERVCC
3566
3567                                  ;ENTERED HERE TO INITIALIZE AND OUTPUT A WORST CASE CHARACTER
```

```
3568                                        ;PATTERN OF '125 & 252'.
3569
3570  013342  112762  000252  024250  WORST:  MOVB    #252,PATERN(R2)
3571  013350  105162  024250          WORST1: COMB    PATERN(R2)          ;COMPLIMENT PATTERN
3572  013354  116204  024250                  MOVB    PATERN(R2),R4
3573  013360  000641                          BR      SERVCC
3574
3575                                        ;DL11 RECEIVER SERVICE ROUTINE
3576
3577  013362  017204  001530          RECVIT: MOV     @RBUFO(R2),R4       ;READ DATA BUFFER
3578  013366  013705  024424                  MOV     START,R5
3579  013372  062705  001604                  ADD     #900.,R5            ;IS BUFFER LIMIT EXCEEDED?
3580  013376  020500                          CMP     R5,R0               ;IS BUFFER LIMIT EXCEEDED?
3581  013400  003021                          BGT     RECVON              ;NO, PROCESS CHARACTER
3582  013402  017737  165716  014020          MOV     @SWR,R3PXT          ;IF SW0 AND SW1 ARE SET -
3583  013410  042737  177774  014020          BIC     #177774,R3PXT       ;DO NOT EXIT- RESET THE BUFFER
3584  013416  022737  000003  014020          CMP     #SW00+SW01,R3PXT    ;AND CONTINUE.
3585  013424  001003                          BNE     1$
3586  013426  013700  024424                  MOV     START,R0
3587  013432  000404                          BR      RECVON
3588  013434  122704  000014          1$:     CMPB    #EOP,R4             ;TRANSFER COMPLETE?
3589  013440  001522                          BEQ     RECVDD              ;YES, SET UP TO EXIT
3590  013442  000207                          RTS     PC                  ;NO, IGNORE THE CHARACTER
3591  013444  005762  024170          RECVON: TST     RECSW0(R2)          ;REC, SW. SET?
3592  013450  001010                          BNE     RECVOA              ;YES, LEGAL REC. INTERRUPT
3593  013452  032777  000001  165644          BIT     #SW00,@SWR          ;IS SW '0' SET?
3594  013460  001004                          BNE     RECVOA              ;YES, HOST SENDING DATA
3595  013462  004737  012412          STERR:  JSR     PC,FLGERR           ;NO, ILLEGAL REC. INTR.
3596  013466  000122                          122                         ;DISPLAY A 'BLINKING R'.
3597  013470  000207                          RTS     PC                  ;EXIT
3598
3599  013472  005704                  RECVOA: TST     R4                  ;REC. ERROR BIT SET?
3600  013474  100021                          BPL     RECVOB              ;NO, VALID CHAR.
3601  013476  032704  040000                  BIT     #40000,R4           ;OVERRUN ERROR?
3602  013502  001404                          BEQ     .+12                ;NO
3603  013504  004737  012412                  JSR     PC,FLGERR           ;YES, DATA OVERRUN ERROR
3604  013510  000117                          117                         ;DISPLAY A 'BLINKING O'
3605  013512  000412                          BR      RECVOB
3606  013514  032704  020000                  BIT     #20000,R4           ;FRAMING ERROR?
3607  013520  001404                          BEQ     .+12                ;NO
3608  013522  004737  012412                  JSR     PC,FLGERR           ;YES, DISPLAY A 'BLINKING F'
3609  013526  000106                          106
3610  013530  000403                          BR      RECVOB
3611  013532  004737  012412                  JSR     PC,FLGERR                   ;PARITY ERROR
3612  013536  000120                          120                         ;DISPLAY A BLINKING 'P'
3613  013540  017737  165560  014020  RECVOB: MOV     @SWR,R3PXT          ;READ THE SWITCHES
3614  013546  042737  177774  014020          BIC     #177774,R3PXT
3615  013554  022737  000003  014020          CMP     #SW00+SW01,R3PXT    ;LOOP DATA BACK REQUESTED
3616  013562  001524                          BEQ     RECVAC              ;YES
3617  013564  105704                          TSTB    R4                  ;= NULL CHAR.?
3618  013566  001513                          BEQ     RECVDE              ;YES, IGNORE IT
3619  013570  005762  024220                  TST     STCODO(R2)          ;HAVE WE RECEIVED THE START CODE?
3620  013574  001014                          BNE     RECVBB              ;YES, SAVE DATA
3621  013576  122704  000377                  CMPB    #377,R4             ;NO, IS THIS IT?
3622  013602  001107                          BNE     RECVAB              ;NO, CHECK 'SW0'
3623  013604  112720  000136                  MOVB    #136,(R0)+          ;YES, DISPLAY ST. CODE AS '↑'
```

```
3624  013610  112720  000012                  MOVB    #12,(R0)+               ;ST NEW LINE
3625  013614  112710  000031                  MOVB    #EOS,(R0)               ;TERMINATE BUFFER.
3626  013620  005262  024220                  INC     STCODO(R2)              ;SET SW.
3627  013624  000474                          BR      RECVOE                  ;EXIT
3628
3629  013626  122704  000014        RECVBB: CMPB    #EOP,R4                 ;TRANSFER DONE?
3630  013632  001425                          BEQ     RECVDD                  ;YES, SET UP TO EXIT
3631  013634  005262  024314                  INC     RECTRO(R2)              ;KEEP TRACK OF NO. OF RECV'D CHAR.S'S
3632  013640  120421                          CMPB    R4,(R1)+                ;COMPARE REC. DATA TO TRANS. DATA
3633  013642  001415                          BEQ     RECVOC                  ;DATA OK
3634  013644  005762  024262        RECVBC: TST     BLKERR(R2)              ;HAS A BLOCK ERROR BEEN DETECTED?
3635  013650  001002                          BNE     1$                      ;YES, DON'T UPDATE ERROR COUNTER
3636  013652  005262  024334                  INC     ERFLGO(R2)              ;DATA ERROR SET SOFT SWITCH
3637  013656  005262  024262        1$:     INC     BLKERR(R2)
3638  013662  112720  000374                  MOVB    #374,(R0)+              ;DISPLAY CHAR IN "BOLD-UNDERLINE"
3639  013666  110420                          MOVB    R4,(R0)+
3640  013670  112720  000360                  MOVB    #360,(R0)+              ;CLR FIELD PRESET
3641  013674  000401                          BR      .+4
3642  013676  110420                  RECVOC: MOVB    R4,(R0)+                ;DISPLAY REC. CHAR.
3643  013700  112710  000031                  MOVB    #EOS,(R0)
3644  013704  000207                          RTS     PC                      ;EXIT
3645  013706  005262  024304          RECVDD: INC     TRANFO(R2)              ;TRANSFER COUNTER
3646  013712  005062  024374                  CLR     BUSYO(R2)
3647  013716  005062  024170                  CLR     RECSWO(R2)              ;CLR. REC. SOFTWARE SW.
3648  013722  005062  024220                  CLR     STCODO(R2)              ;CLR ST. CODE SW.
3649  013726  005762  024364                  TST     XFERO(R2)               ;IS AUTO TRANSFER SW. SET?
3650  013732  001431                          BEQ     RECVOE                  ;NO, NORMAL TRANSFER EXIT
3651  013734  005762  024324                  TST     STOPO(R2)               ;YES, STOP SW. SET?
3652  013740  001402                          BEQ     .+6                     ;NO, START NEXT TRANSFER
3653  013742  000137  012556                  JMP     RELOAD                  ;YES, RESTART
3654  013746  016200  024354                  MOV     RESTRO(R2),R0           ;NO, RESET BUFFER POINTER
3655  013752  005762  024334                  TST     ERFLGO(R2)              ;ERROR FLAG SET?
3656  013756  001412                          BEQ     RECVDD                  ;NO, START NEXT TRANSFER
3657  013760  005777  165340          TESTSW: TST     @SWR                    ;YES, IS SW15 SET?
3658  013764  100411                          BMI     RECVEE                  ;YES, INHIBIT FURTHER TRANSFERS
3659  013766  005762  024240                  TST     HOLDSO(R2)              ;HAS SW15 BEEN SET AND NOW RESET?
3660  013772  001404                          BEQ     RECVDD                  ;NO, IT HASN'T BEEN SET
3661  013774  005062  024334                  CLR     ERFLGO(R2)              ;YES, RESET THE ERROR COUNTER TO '0'
3662  014000  005062  024240                  CLR     HOLDSO(R2)
3663  014004  000137  012770          RECVDD: JMP     SERVBB                  ;SET UP NEXT TRANSFER
3664  014010  012762  000001  024240  RECVEE: MOV     #1,HOLDSO(R2)           ;SET TO HOLD XFER
3665  014016  000207                  RECVOE: RTS     PC
3666  014020  000000                  R3PXT:  .WORD   0
3667
3668
3669  014022  032777  000001  165274  RECVAB: BIT     #SW00,@SWR              ;IS SW '0' SET?
3670  014030  001322                          BNE     RECVOC                  ;YES, JUST DISPLAY CHARACTER.
3671  014032  000704                          BR      RECVBC                  ;NO, POST AS ILLEGAL CHAR.
3672
3673                                  ;ENTERED HERE IF BOTH 'SW00 & SW01' ARE SET
3674
3675  014034  005262  024314          RECVAC: INC     RECTRO(R2)              ;UPDATE RECEIVE COUNT
3676  014040  005262  024304                  INC     TRANFO(R2)              ;UPDATE TRANSMIT COUNT
3677  014044  005262  024200                  INC     TRNSWO(R2)              ;SET THE TRANSMITTER SWITCH
3678  014050  110472  001570                  MOVB    R4,@XBUFO(R2)           ;TRANSMIT CHARACTER BACK
3679  014054  042704  177400                  BIC     #177400,R4              ;SET UP TO DISPLAY OCTAL REPRESENTATION
```

```
3680                                                                    ;OF THE CHARACTER
3681  014060  012737  000003  024470          MOV     #3,KSTOR5
3682  014066  012737  000376  015166          MOV     #376,MASK
3683  014074  012703  000006                  MOV     #6,R3
3684  014100  000401                          BR      .+4
3685  014102  006104                  MOVE1:  ROL     R4
3686  014104  006104                          ROL     R4
3687  014106  006104                          ROL     R4
3688  014110  005337  024470                  DEC     KSTOR5
3689  014114  002010                          BGE     MOVE2
3690  014116  110402                          MOVB    R4,R2
3691  014120  143702  015166                  BICB    MASK,R2
3692  014124  152702  000060                  BISB    #60,R2
3693  014130  110220                          MOVB    R2,(R0)+                ;PUT CHARACTER IN DISPLAY BUFFER
3694  014132  112710  000031                  MOVB    #EOS,(R0)               ;TERMINATE BUFFER
3695  014136  012737  000370  015166  MOVE2:  MOV     #370,MASK
3696  014144  005303                          DEC     R3                      ;DONE
3697  014146  001355                          BNE     MOVE1                   ;NO
3698  014150  112720  000040                  MOVB    #40,(R0)+               ;YES, PLACE 'SPACE' CODE IN BUFFER
3699  014154  112710  000031                  MOVB    #EOS,(R0)
3700  014160  000207                          RTS     PC                      ;RETURN
3701                                  ;*************************************************************************
3702                                  ;ROUTINE TO LOOP THRU A SINGLE LOGIC SUBTEST.  ENTERED FROM THE 'MONITOR'
3703                                  ;VIA SELECTING TEST '?'.  UPON ENTERING THIS SUBROUTINE ANOTHER 'HALT' IS EN-
3704                                  ;COUNTERED.  LOAD THE ADDRESS OF THE SUBTEST TO BE EXECUTED IN THE DATA
3705                                  ;SWITCHES AND PRESS CONTINUE.
3706                                  ;NOTE THAT 'SW11' MUST BE '0' (DOWN) TO RUN THIS TEST.
3707                                  ;*************************************************************************
3708
3709  014162  005737  024154  TESTX:  TST     TTYSWH                  ;TTY AVAILABLE?
3710  014166  001406                  BEQ     TSTA                    ;NO, HALT FOR TEST ADDR.
3711  014170  104012                  PRINT
3712  014172  021634                  MESB                            ;TEXT 'TEST ADDR.? '
3713  014174  104015                  ASEMBL                          ;YES, GET ADDR. FROM TTY.
3714  014176  010337  024460          MOV     R3,KSTOR1               ;SAVE ADDRESS
3715  014202  000404                  BR      TSTB                    ;EXECUTE TEST
3716  014204  000000          TSTA:   HALT                            ;GET TEST ADDRESS
3717  014206  017737  165112  024460  MOV     @SWR,KSTOR1             ;SAVE ADDRESS
3718
3719  014214  023727  024460  006174  TSTB:   CMP     KSTOR1,#SETUPF  ;'FK' OR 'VT' LOGIC TEST?
3720  014222  101003                  BHI     .+10                    ;SET UP FOR 'FK'
3721  014224  004737  003026          JSR     PC,SETUPV               ;SET UP FOR 'VT'
3722  014230  000402                  BR      .+6
3723  014232  004737  006174          JSR     PC,SETUPF
3724  014236  062737  000002  024460  ADD     #2,KSTOR1               ;ADD '2' TO POINT TO INSTRUCTION AFTER SCOPE
3725  014244  017737  010210  024142  MOV     @KSTOR1,TSTNUM          ;LOAD TEST NO.
3726  014252  062737  000002  024460  ADD     #2,KSTOR1
3727  014260  005037  020404  XLOOP:  CLR     SCOPEF                  ;KEEP COUNT AT ZERO
3728  014264  012737  014260  020406  MOV     #XLOOP,RETURN           ;LOAD SCOPE LOOP RETURN POINTER
3729  014272  000177  010162          JMP     @KSTOR1                 ;JUMP TO TEST
3730                                  ;*************************************************************************
3731                                  ;SUBROUTINE TO ISSUE N SPACES
3732                                  ;N IS ONE PLUS VALUE CONTAINED IN SPACEX
3733                                  ;SPACEX IS CLEARED WITHIN THE SUBROUTINE, SO THAT A CALL ON
3734                                  ;SPACE WITHOUT LOADING SPACEX ISSUES ONLY ONE SPACE
3735                                  ;*************************************************************************
```

```
3736
3737  014276  105777  165016          XSPACE: TSTB    @TPS            ;WAIT FOR TTY READY
3738  014302  100375                          BPL     .-4
3739  014304  012777  000240  165010          MOV     #240,@TPB       ;OUTPUT A SPACE
3740  014312  005337  014326                  DEC     SPACEX          ;DECREMENT COUNT
3741  014316  003367                          BGT     XSPACE          ;LOOP IF NOT DONE
3742  014320  005037  014326                  CLR     SPACEX          ;RESET COUNT TO ZERO
3743  014324  000002                          RTI                     ;RETURN
3744  014326  000000                  SPACEX: 0
3745                                  ;*************************************************************************
3746                                  ;SUBROUTINE TO TEST FOR THE KEYBOARD FLAG BEING SET
3747                                  ;*************************************************************************
3748
3749  014330  005737  024154          TKSFLG: TST     TTYSWH          ;USING THE 'TTY' ?
3750  014334  001404                          BEQ     .+12            ;NO, EXIT
3751  014336  105777  164752                  TSTB    @TKS            ;FLAG SET?
3752  014342  100001                          BPL     .+4             ;NO, EXIT
3753  014344  104013                          TTYIN                   ;YES,INQUIRE
3754  014346  000002                          RTI
3755                                  ;*************************************************************************
3756                                  ;KEYBOARD SERVICE ROUTINE
3757                                  ;*************************************************************************
3758  014350  012704  014600          XTTYIN: MOV     #INBUF,R4       ;SETUP CHARACTER BUFFER
3759  014354  005037  024146                  CLR     CHRCNT          ;CLEAR CHARACTER COUNTER
3760  014360  005037  014602                  CLR     INBUF+2
3761  014364  105777  164724          INPUTA: TSTB    @TKS            ;CHARACTER READY?
3762  014370  100375                          BPL     INPUTA          ;NO, WAIT IT OUT
3763  014372  017701  164720                  MOV     @TKB,R1         ;SAVE CHARACTER
3764  014376  042701  000200                  BIC     #200,R1         ;STRIPE PARITY BIT
3765  014402  120127  000060                  CMPB    R1,#60          ;IS IT A SPECIAL CHARACTER
3766  014406  100420                          BMI     SPCHR1          ;YES, TEST IT
3767  014410  122701  000137                  CMPB    #137,R1
3768  014414  100415                          BMI     SPCHR1
3769  014416  010124                  INPUTB: MOV     R1,(R4)+        ;SAVE CHARACTER
3770  014420  005237  024146                  INC     CHRCNT          ;INCREMENT THE CHARACTER COUNT.
3771  014424  022737  000016  024146          CMP     #14.,CHRCNT
3772  014432  100450                          BMI     SPCHR5          ;TYPE '?' IF TOO MANT CHAR.
3773  014434  105777  164660          OUTPTA: TSTB    @TPS            ;ECHO CHARACTER
3774  014440  100375                          BPL     OUTPTA
3775  014442  110177  164654                  MOVB    R1,@TPB
3776  014446  000746                          BR      INPUTA          ;WAIT FOR NEXT CHARACTER
3777
3778                                  ;SUBROUTINE TO TEST FOR SPECIAL CHARACTERS :'↑C','+','CR','.' OR 'RUBOUT'
3779  014450  122701  000003          SPCHR1: CMPB    #3,R1           ;CHAR.='↑C'
3780  014454  001002                          BNE     .+6             ;NO,NOT '↑C'
3781  014456  000137  002526                  JMP     MONITR          ;YES, EXIT TO MONITOR
3782  014462  122701  000001                  CMPB    #1,R1           ;CHAR. ='↑A'?
3783  014466  001005                          BNE     .+14            ;NOT '↑A'
3784  014470  022626                          POP2SP                  ;YES, RESTORE STACK
3785  014472  104012                          PRINT
3786  014474  023001                          CNTRLA                  ;TEXT '↑A'
3787  014476  000177  007426                  JMP     @AVECTR
3788  014502  122701  000177                  CMPB    #177,R1         ;CHAR. = 'RUBOUT'
3789  014506  001011                          BNE     SPCHR3          ;IGNORE CHAR. & EXIT
3790  014510  005737  024146                  TST     CHRCNT          ;IS RUBOUT LEGAL?
3791  014514  001723                          BEQ     INPUTA          ;NO, IGNORE IT
```

```
3792  014516  005337  024146                  DEC     CHRCNT
3793  014522  012701  000134                  MOV     #134,R1         ;TYPE '\' TO INDICATE RUBOUUT
3794  014526  005744                          TST     -(R4)           ;POP OFF LAST CHARACTER
3795  014530  000741                          BR      OUTPTA          ;WAIT FOR NEXT CHARACTER
3796  014532  122701  000054          SPCHR3: CMPB    #54,R1          ;TEST FOR ','
3797  014536  001727                          BEQ     INPUTB          ;LEGAL CHAR., SAVE IT
3798  014540  122701  000015          SPCHR4: CMPB    #15,R1          ;=TO 'CARRIAGE RETURN' TO TERMINATE?
3799  014544  001003                          BNE     SPCHR5          ;NO, CONTINUE
3800  014546  104012                          PRINT                   ;YES, TYPE 'CR-LF'
3801  014550  023005                          CRLF
3802  014552  000002          EXTTY:  RTI                             ;EXIT
3803  014554  122701  000040          SPCHR5: CMPB    #40,R1          ;TEST FOR SPACE
3804  014560  001725                          BEQ     OUTPTA          ;ECHO BUT DON'T SAVE
3805  014562  105701                          TSTB    R1              ;TEST FOR A NULL CHAR.
3806  014564  001002                          BNE     .+6             ;NO, EHCO IT
3807  014566  000137  002526                  JMP     MONITR          ;ILLEGAL CHAR. RETURN TO MONITR
3808  014572  104012                          PRINT                   ;OTHERWISE TYPE '?'
3809  014574  023007                          QMARK
3810  014576  0C0664                          BR XTTYIN               ;WAIT FOR NEW ENTRY
3811  014600  000000          INBUF:  0                               ;CHARACTER STORAGE BUFFER
3812          014632                  .=.+30
3813                          ;*****************************************************************
3814                          ;THIS ROUTINE CHECKS THE 'SWAP SW.' AND IF FOUND SET, SETS UP
3815                          ;THE OPPSITE UNIT TO BE TESTED.
3816                          ;*****************************************************************
3817
3818  014632  005037  024162  TSWAP:  CLR     DONESW
3819  014636  005737  024164          TST     SWAPEM          ;SWAP SW. SET?
3820  014642  001001                  BNE     .+4             ;YES, SWITCH UNITS
3821  014644  000207                  RTS     PC              ;NO, EXIT
3822  014646  005077  164636          CLR     @VTCSR          ;CLR GO BIT
3823  014652  005237  024134          INC     UNITFG          ;SELECT OPPOSITE UNIT
3824  014656  004737  016320          JSR     PC,STUNIT       ;SET IT UP
3825  014662  005737  024134          TST     UNITFG          ;BACK TO '0'
3826  014666  001414                  BEQ     SWPEXT          ;YES, CONTINUE TO NEXT TEST
3827  014670  005726                  POP1SP                  ;CLEAN UP STACK
3828  014672  104012                  PRINT
3829  014674  023731                  MES44                   ;TEXT 'RUNNING UNIT 'X'
3830  014676  112737  000260  014750          MOVB    #260,   MRUNN1
3831  014704  153737  024134  014750          BISB    UNITFG, MRUNN1
3832  014712  104012                  PRINT
3833  014714  014750                  MRUNN1
3834  014716  000002                  RTI                     ;RE-RUN TEST W/ UNIT 'X'
3835  014720  104012          SWPEXT: PRINT
3836  014722  023711                  MES43                   ;TEXT 'RUNNING UNIT '0'
3837  014724  005737  024240          TST     HOLDSO
3838  014730  001406                  BEQ     .+16
3839  014732  032777  010000  164364          BIT     #SW12,@SWR
3840  014740  001002                  BNE     .+6
3841  014742  005726                  POP1SP
3842  014744  000002                  RTI
3843  014746  000207                  RTS     PC
3844  014750     000     015     012  MRUNN1: .BYTE   0,15,12,100
3845  014753     100
3846
3847                          ;*****************************************************************
```

```
3848                                    ;MESSAGE PRINT ROUTINE, ENTERED VIA EMT DISPATCH HANDLER.
3849                                    ;ROUTINE PICKS UP CONTENTS OF THE 'PC' AND USES THIS AS
3850                                    ;THE ADDRESS OF MESSAGE TO BE TYPED.
3851                                    ;*****************************************************************************
3852
3853  014754  004737  017062       TYPMES: JSR     PC,TTYENB       ;ENABLE TTY INTERRUPTS
3854  014760  104002                       SAVREG                  ;SAVE REGISTERS ON STACK
3855  014762  017602  000000               MOV     @(SP),R2        ;GET THE MESSAGE ADDRESS FROM START
3856  014766  062716  000002               ADD     #2,(SP)         ;SET UP STACK TO EXIT
3857  014772  105777  164322       TYPERA: TSTB    @TPS
3858  014776  100375                       BPL     TYPERA          ;WAIT FOR TTY DONE
3859  015000  122712  000100               CMPB    #100,(R2)       ;TEST FOR '@'
3860  015004  001003                       BNE     TYPER1          ;BRANCH IF NO EQUAL
3861  015006  104003                       GETREG                  ;RESTORE REGISTERS FROM STACK.
3862  015010  104022                       NULL                    ;TRANSMIT 'NULL' CHARACTER
3863  015012  000002                       RTI                     ;OTHERWISE EXIT
3864  015014  122712  000045       TYPER1: CMPB    #45,(R2)        ;TEST FOR '%'
3865  015020  001403                       BEQ     TYPECL          ;IF = TYPE 'CR-_F'
3866  015022  112277  164274       TYPER2: MOVB    (R2)+,@TPB      ;OUTPUT CHAR.
3867  015026  000761                       BR      TYPERA
3868  015030  012777  000015  164264 TYPECL: MOV     #15,@TPB        ;TYPE 'CR'
3869  015036  105777  164256               TSTB    @TPS
3870  015042  100375                       BPL     .-4
3871  015044  012777  000012  164250       MOV     #12,@TPB
3872  015052  105722                       TSTB    (R2)+           ;INCREMENT BUFFER
3873  015054  000746                       BR      TYPERA
3874                                    ;*****************************************************************************
3875                                    ;SUBROUTINE TO TYPEOUT A '6' DIGIT OCTAL NO. THE 'PC' CONTAINS
3876                                    ;THE ADDRESS OF 'WORD' TO BE TYPED
3877                                    ;*****************************************************************************
3878
3879  015056  004737  017062       OCTPRT: JSR     PC,TTYENB       ;ENABLE TTY INTERRUPTS
3880  015062  104002                       SAVREG                  ;SAVE REGISTERS ON STACK
3881  015064  017601  000000               MOV     @(SP),R1        ;THE ADDRESS OF WORD TO BE TYPED
3882  015070  062716  000002               ADD     #2,(SP)         ;SET UP STACK TO EXIT
3883  015074  012737  000006  024466       MOV     #6,KSTOR4
3884  015102  012737  000376  015166       MOV     #376,MASK       ;MASK FOR FIRST BIT
3885  015110  000401                       BR      .+4
3886  015112  006111               MOVEIT: ROL     (R1)
3887  015114  006111                       ROL     (R1)
3888  015116  006111                       ROL     (R1)
3889  015120  111102                       MOVB    (R1),R2
3890  015122  143702  015166               BICB    MASK,R2
3891  015126  052702  000260               BIS     #260,R2
3892  015132  132777  000200  164160       BITB    #200,@TPS
3893  015140  100374                       BPL     .-6
3894  015142  110277  164154               MOVB    R2,@TPB         ;PRINT CHAR.
3895  015146  012737  000370  015166       MOV     #370,MASK       ;MASK FOR NEXT '5' DIGITS
3896  015154  005337  024466               DEC     KSTOR4
3897  015160  001354                       BNE     MOVEIT
3898  015162  104003                       GETREG                  ;RESTORE REGISTERS FROM STACK.
3899  015164  000002                       RTI
3900  015166  000376               MASK:   376
3901                                    ;*****************************************************************************
3902                                    ;THIS SUBROUTINE IS USED TO CREATE THE 'VT' CHARACTER SET IN NUMERICAL ORDER
3903                                    ;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  024152        INTCHR: TST     INTSWH                  ;INITIALIZE SUBROUTINE?
015174  001014                        BNE     LDCHAR                  ;NO,LOAD CHAR.
015176  012700  024474                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  024152                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
015402  001416                        BEQ     2$
015404  120127  000213                CMPB    R1,#213
015410  001413                        BEQ     2$
015412  120127  000215                CMPB    R1,#215
```

```
3960  015416  001410                        BEQ     2$
3961  015420  120127  000217                CMPB    R1,#217
3962  015424  001405                        BEQ     2$
3963  015426  120127  000255                CMPB    R1,#255
3964  015432  001402                        BEQ     2$
3965  015434  005137  001622                COM     FCSET1
3966  015440  120127  000212        2$:       CMPB    R1,#VISEOL              ;CHAR.=VISIBLE END OF LINE?
3967  015444  001407                        BEQ     LDCHR2                  ;YES, SKIP IT
3968  015446  120127  000214                CMPB    R1,#VISEOP              ;CHAR.=VISIBLE END OF PARAGRAPH?
3969  015452  001404                        BEQ     LDCHR2                  ;YES, SKIP IT
3970  015454  120127  000231                CMPB    R1,#VISEOS              ;CHAR.=VISIBLE END OF SCREEN?
3971  015460  001401                        BEQ     .+4                     ;NO,
3972  015462  000207                        RTS     PC
3973  015464  005201                LDCHR2: INC     R1                      ;YES, SKIP IT
3974  015466  005037  001622                CLR     FCSET1
3975  015472  000137  015304                JMP     LDCHEB                  ;RETJPN TO 'CALL+2'
3976
3977                                        ;*************************************************************************
3978                                        ;SUBROUTINE TO SET UP A WAIT LOOP TO WAIT FOR VERTICAL SYNC (17 MSEC.).
3979                                        ;*************************************************************************
3980
3981  015476  004737  017062        XWAITS: JSR     PC,TTYENB               ;ENABLE TTY INTERRUPTS
3982  015502  012737  170000  024442        MOV     #-10000,TEMP2
3983  015510  005237  024442                INC     TEMP2
3984  015514  001375                        BNE     .-4
3985  015516  000002                        RTI
3986
3987                                        ;FOLLOWING IS A '16' BYTE BUFFER USED TO TERMINATE EACH LINE OF THE 'VT' BUFFER.
3988
3989  015520      212               TERMTB: .BYTE   VISEOL
3990  015521      214                       .BYTE   VISEOP
3991  015522      212                       .BYTE   VISEOL
3992  015523      214                       .BYTE   VISEOP
3993  015524      212                       .BYTE   VISEOL
3994  015525      214                       .BYTE   VISEOP
3995  015526      212                       .BYTE   VISEOL
3996  015527      214                       .BYTE   VISEOP
3997  015530      212                       .BYTE   VISEOL
3998  015531      214                       .BYTE   VISEOP
3999  015532      212                       .BYTE   VISEOL
4000  015533      214                       .BYTE   VISEOP
4001  015534      212                       .BYTE   VISEOL
4002  015535      214                       .BYTE   VISEOP
4003  015536      212                       .BYTE   VISEOL
4004  015537      231                       .BYTE   VISEOS
4005                                        .EVEN
4006
4007                                        ;*************************************************************************
4008                                        ;SUBROUTINE TO SET UP AN APPROXIMATE '3' SECOND DELAY. THE ROUTINE ALSO
4009                                        ;CONTINUOUSLY MONITORS DATA 'SW14' AND IF FOUND SET CAUSES AN INDEFINITE DELAY.
4010                                        ;*************************************************************************
4011
4012  015540  012737  175000  024442 XDELAY: MOV     #175000,TEMP2           ;SET UP SHORT DELAY
4013  015546  000402                        BR      .+6
4014  015550  005037  024442        XDLAYL: CLR     TEMP2                   ;SET UP LONG DELAY
4015  015554  004737  017062                JSR     PC,TTYENB               ;ENABLE TTY INTERRUPTS
```

```
4016  015560  012737  177777  024440          MOV     #-1,TEMP1
4017  015566  005737  024426          XDLAY1: TST     DISPSW                  ;DISPLAYING MESSAGE?
4018  015572  001010                          BNE     XDLAY3                  ;YES INHIBIT HOLD
4019  015574  032777  040000  163522          BIT     #SW14,@SWR              ;TEST DATA SW14
4020  015602  001374                          BNE     .-6                     ;IF SET, WAIT FOR CLEAR.
4021  015604  032777  004000  163512          BIT     #SW11,@SWR              ;IS SW. '11' SET?
4022  015612  001006                          BNE     XDLAY2                  ;YES, INIBIT DELAY
4023  015614  005237  024442          XDLAY3: INC     TEMP2
4024  015620  001362                          BNE     XDLAY1
4025  015622  005237  024440                  INC     TEMP1
4026  015626  001357                          BNE     XDLAY1
4027  015630  000002                  XDLAY2: RTI
4028                                  ;*************************************************************************
4029                                  ;SUBROUTINE TO LOAD '64' CHAR.'S INTO 'VT' BUFFER.
4030                                  ;*************************************************************************
4031
4032  015632  104002                  XLDLNE: SAVREG                          ;SAVE REG.'S
4033  015634  012701  177700                  MOV     #-64.,R1                ;LOAD '64' CHAR. BYTES
4034  015640  013702  024460                  MOV     KSTOR1,R2               ;FROM THIS ADDR. UP.
4035  015644  112220                  XLDLNA: MOVB    (R2)+,(R0)+             ;'R0' IS POINTER TO A 'VT' BUFFER ADDR.
4036  015646  005201                          INC     R1                      ;FINISHED?
4037  015650  001375                          BNE     XLDLNA                  ;NO, LOAD NEXT CHAR.
4038  015652  104003                          GETREG                          ;YES, RESTORE REG.'S
4039  015654  000002                          RTI
4040
4041                                  ;*************************************************************************
4042                                  ;SUBROUTINE TO LOAD THE 'VT' BUFFER W/ 'VISIBLE EOS' CHARACTER.
4043                                  ;*************************************************************************
4044
4045  015656  104002                  XEOSBF: SAVREG                          ;SAVE REG.'S
4046  015660  012700  024474                  MOV     #VTBUFF,R0              ;SET UP BUFFER ADDR. POINTER
4047  015664  012701  004374                  MOV     #2300.,R1               ;LOAD '515' DATA WORDS
4048  015670  112720  000031                  MOVB    #EOS,(R0)+
4049  015674  005301                          DEC     R1
4050  015676  001374                          BNE     .-6
4051  015700  012700  024474                  MOV     #VTBUFF,R0              ;RE-SET POINTER
4052  015704  005077  163574                  CLR     @VTCAR                  ;CLR CURSOR ADDR. REG.
4053  015710  010077  163572                  MOV     R0,@VTSAR               ;LOAD STARTING ADDR.
4054  015714  012777  000001  163566          MOV     #1,@VTCSR               ;ST. DISPLAY
4055  015722  104003                          GETREG
4056  015724  000002                          RTI
4057
4058                                  ;*************************************************************************
4059                                  ;SUBROUTINE ENTERED AT THE END OF EVERY DISPLAY TEST TO CHECK STATUS
4060                                  ;OF THE DATA SWITCHES
4061                                  ;*************************************************************************
4062
4063  015726  005237  024162          XTHEND: INC     DONESW
4064  015732  104020                  XENDT:  DELAYL                          ;ISSUE DELAY
4065  015734  004737  017062                  JSR     PC,TTYENB               ;ENABLE TTY INTERRUPTS
4066  015740  005737  024150                  TST     RUNSWH                  ;RUNING ALL TESTS?
4067  015744  001015                          BNE     RUNEM                   ;YES, DO IT
4068  015746  032777  010000  163350          BIT     #SW12,@SWR              ;MANUAL ADVANCE SW. SET?
4069  015754  001011                          BNE     RUNEM                   ;YES,
4070  015756  013737  024162  024240          MOV     DONESW,HOLDSO           ;SAVE SW.
4071  015764  004737  014632                  JSR     PC,TSWAP                ;TEST FOR SWAP SW.
```

```
4072  015770  005737  024240                TST     HOLDSO                  ;WAS SW. SET TO LOOP TEST?
4073  015774  001354                        BNE     XTHEND                  ;YES, LOOP IT.
4074  015776  000002                        RTI                             ;NORMAL EXIT
4075  016000  104020              RUNEM:    DELAYL
4076  016002  004737  014632                JSR     PC,TSWAP                ;TEST SWAP SW.
4077  016006  062716  000002      RUNEXT:   ADD     #2,(SP)                 ;ADVANCE TO NEXT TEST
4078  016012  000002                        RTI
4079
4080                                ;*************************************************************************
4081                                ;POWER FAIL HANDLER
4082                                ;*************************************************************************
4083
4084  016014  010046              PWRFAL:   MOV     R0,-(SP)          ;SAVE REGISTERS ON STACK
4085  016016  010146                        MOV     R1,-(SP)
4086  016020  010246                        MOV     R2,-(SP)
4087  016022  010346                        MOV     R3,-(SP)
4088  016024  010446                        MOV     R4,-(SP)
4089  016026  010546                        MOV     R5,-(SP)
4090  016030  013746  000024                MOV     24,-(SP)
4091  016034  010637  024440                MOV     SP,TEMP1          ;SAVE STACK POINTER
4092  016040  012737  016050  000024        MOV     #PWRUP,@#24       ;POWER UP ROUTINE TO LOCATION 24
4093  016046  000000                        HALT                      ;HALT
4094  016050  012777  000340  163234  PWRUP:  MOV     #340,@PSW         ;LOCK OUT INTERRUPTS
4095  016056  013706  024440                MOV     TEMP1,SP          ;POWER DOWN ROUTINE TO LOCATION 24
4096  016062  012637  000024                MOV     (SP)+,@#24
4097  016066  012605                        MOV     (SP)+,R5          ;RESTORE REGISTERS
4098  016070  012604                        MOV     (SP)+,R4
4099  016072  012603                        MOV     (SP)+,R3
4100  016074  012602                        MOV     (SP)+,R2
4101  016076  012601                        MOV     (SP)+,R1
4102  016100  012600                        MOV     (SP)+,R0
4103  016102  000005                        RESET
4104  016104  005001                        CLR     R1                ;POWER UP DELAY
4105  016106  005201                        INC     R1
4106  016110  001376                        BNE     .-2
4107  016112  104000                        DISPLAY
4108  016114  023037                        MES21
4109  016116  000177  002264                JMP     @RETURN
4110
4111                                ;*************************************************************************
4112                                ;SUBROUTINE TO SAVE 'R1-R5' ON STACK
4113                                ;*************************************************************************
4114
4115  016122  012637  024450      XSAVRG:   MOV     (SP)+,SAVEPC
4116  016126  012637  024452                MOV     (SP)+,SAVPSW
4117  016132  012637  024454                MOV     (SP)+,SAV2PC
4118  016136  012637  024456                MOV     (SP)+,SAV2SW
4119  016142  010146                        MOV     R1,-(SP)
4120  016144  010246                        MOV     R2,-(SP)
4121  016146  010346                        MOV     R3,-(SP)
4122  016150  010446                        MOV     R4,-(SP)
4123  016152  010546                        MOV     R5,-(SP)
4124  016154  013746  024456                MOV     SAV2SW,-(SP)
4125  016160  013746  024454                MOV     SAV2PC,-(SP)
4126  016164  013746  024452                MOV     SAVPSW,-(SP)
4127  016170  013746  024450                MOV     SAVEPC,-(SP)
```

```
 4128  016174  000002                        RTI
 4129                          ;*************************************************************************
 4130                          ;SUBROUTINE TO RESTORE 'R1-R5' FROM THE STACK
 4131                          ;*************************************************************************
 4132
 4133  016176  012637  024450  XGETRG: MOV     (SP)+,SAVEPC
 4134  016202  012637  024452         MOV     (SP)+,SAVPSW
 4135  016206  012637  024454         MOV     (SP)+,SAV2PC
 4136  016212  012637  024456         MOV     (SP)+,SAV2SW
 4137  016216  012605                 MOV     (SP)+,R5
 4138  016220  012604                 MOV     (SP)+,R4
 4139  016222  012603                 MOV     (SP)+,R3
 4140  016224  012602                 MOV     (SP)+,R2
 4141  016226  012601                 MOV     (SP)+,R1
 4142  016230  013746  024456         MOV     SAV2SW,-(SP)
 4143  016234  013746  024454         MOV     SAV2PC,-(SP)
 4144  016240  013746  024452         MOV     SAVPSW,-(SP)
 4145  016244  013746  024450         MOV     SAVEPC,-(SP)
 4146  016250  000002                 RTI
 4147
 4148                          ;*************************************************************************
 4149                          ;SUBROUTINE TO PRE-LOAD THE 'VT' DATA BUFFER WITH DATA FROM THE 'PC'.
 4150                          ;*************************************************************************
 4151
 4152  016252  005077  163232  XPRELD: CLR     @VTCSR          ;STOP 'VT'
 4153  016256  012701  001000         MOV     #512.,R1        ;SET UP '512' WORDS
 4154  016262  011602                 MOV     (SP),R2         ;GET ADDR. OF DATA FROM STACK
 4155  016264  062716  000002         ADD     #2,(SP)         ;SET UP STACK TO EXIT
 4156  016270  012700  024474         MOV     #VTBUFF,R0      ;SET UP BUFFER ADDR. POINTER
 4157  016274  011220                 MOV     (R2),(R0)+      ;MOVE DATA TO BUFFER.
 4158  016276  005301                 DEC     R1              ;DONE?
 4159  016300  001375                 BNE     .-4             ;NO, MOVE NEXT WORD
 4160  016302  112710  014431         MOVB    #14431,(R0)     ;YES, TERMINATE BUFFER W/ 'EOS'
 4161  016306  012700  024474         MOV     #VTBUFF,R0      ;RE-SET BUFFER ADDR. POINTER
 4162  016312  010077  163170         MOV     R0,@VTSAR       ;LOAD 'STARTING ADDR. REG.
 4163  016316  000002                 RTI                     ;EXIT
 4164
 4165                          ;*************************************************************************
 4166                          ;SUBROUTINE TO SET UP THE 'FK' AND THE 'VT' DEVICE ADDRESSES
 4167                          ;*************************************************************************
 4168
 4169  016320  123737  024134  024136  STUNIT: CMPB    UNITFG,UNITNO           ;TESTED ALL UNITS?
 4170  016326  001414                 BEQ     SETUT0          ;YES, RE-SELECT UNIT '0'
 4171  016330  022737  000001  024134         CMP     #1,UNITFG       ;RUNNING UNIT #1?
 4172  016336  001417                 BEQ     SETUT1          ;YES
 4173  016340  022737  000002  024134         CMP     #2,UNITFG       ;RUNNING UNIT #2
 4174  016346  001420                 BEQ     SETUT2
 4175  016350  022737  000003  024134         CMP     #3,UNITFG       ;RUNNING UNIT #3
 4176  016356  001421                 BEQ     SETUT3
 4177  016360  005037  024134  SETJT0: CLR     UNITFG
 4178  016364  012700  001330         MOV     #FK0LDB,R0      ;NO, LOAD UNIT '0'
 4179  016370  012701  001410         MOV     #VT0CAR,R1
 4180  016374  000416                 BR      LDUNIT
 4181  016376  012700  001344  SETUT1: MOV     #FK1LDB,R0
 4182  016402  012701  001424         MOV     #VT1CAR,R1
 4183  016406  000411                 BR      LDUNIT
```

```
 4184  016410  012700  001360          SETUT2: MOV     #FK2LDB,R0
 4185  016414  012701  001440                  MOV     #VT2CAR,R1
 4186  016420  000404                          BR      LDUNIT
 4187  016422  012700  001374          SETUT3: MOV     #FK3LDB,R0
 4188  016426  012701  001454                  MOV     #VT3CAR,R1
 4189  016432  012702  001470          LDUNIT: MOV     #FKLDB,R2
 4190  016436  012703  001504                  MOV     #VTCAR,R3
 4191  016442  012022                  SETFKA: MOV     (R0)+,(R2)+
 4192  016444  012123                          MOV     (R1)+,(R3)+
 4193  016446  022702  001504                  CMP     #FKLVL+2,R2
 4194  016452  001373                          BNE     SETFKA
 4195  016454  000207                          RTS     PC
 4196                          ;*****************************************************************************
 4197                          ;SUBROUTINE ENTERED ON AN ILLEGAL TRAP. THE ROTUINE REPORTS WHERE IT
 4198                          ;TRAPPED 'FROM' AND WHERE IT TRAPPED 'TO'. IF NO TTY IS AVAILABLE
 4199                          ;A 'HALT' IS EXECUTED. HERE THE USER CAN EXAMINE THE 'FROMPC'
 4200                          ;LOCATION AND THE 'TOPC' LOCATION TO TRACK DOWN THE ERROR.
 4201                          ;*****************************************************************************
 4202
 4203  016456  000005          ERTRAP: RESET                   ;CLEAR ALL FLAGS
 4204  016460  104022                  NULL
 4205  016462  000240                  NOP
 4206  016464  011637  024156          MOV     (SP),TOPC       ;SAVE LOCATION WHERE IT TRAPPED 'TO'
 4207  016470  022626                  POP2SP
 4208  016472  011637  024160          MOV     (SP),FROMPC     ;SAVE WHERE IT TRAPPED FROM
 4209  016476  162737  000004  024156  SUB     #4,TOPC
 4210  016504  162737  000002  024160  SUB     #2,FROMPC
 4211  016512  005737  024154          TST     TTYSWH          ;IS TTY AVAILABLE?
 4212  016516  001001                  BNE     .+4             ;YES
 4213  016520  000000                  HALT                    ;NO, HALT
 4214  016522  104012                  PRINT
 4215  016524  022126                  MESSA                   ;TEXT 'ILLEGAL TRAP'
 4216  016526  104014                  PRTOCT
 4217  016530  024156                  TOPC                    ;TYPE 'PC TRAPPED TOO
 4218  016532  104012                  PRINT
 4219  016534  022150                  MES6C                   ;TEXT 'FROM'
 4220  016536  104014                  PRTOCT
 4221  016540  024160                  FROMPC                  ;WHERE IT TRAPPED FROM
 4222  016542  000137  002526          JMP     MONITR          ;RETURN TO THE MONITOR
 4223
 4224                          ;*****************************************************************************
 4225                          ;SUBROUTINE TO PRINT THE DECIMAL VALUE IN R2
 4226                          ;*****************************************************************************
 4227
 4228  016546  005077  162540  XBINDEC: CLR    @PSW            ;ENABLE INTERRUPTS
 4229  016552  042702  160000          BIC     #160000,R2
 4230  016556  012737  177774  016652  MOV     #-4,R3PBT
 4231  016564  012704  016642          MOV     #DECPNT,R4
 4232  016570  012701  177777  TYPT1:  MOV     #-1,R1
 4233  016574  005201          TYPT2:  INC     R1
 4234  016576  161402                  SUB     (R4),R2
 4235  016600  100375                  BPL     TYPT2
 4236  016602  062402                  ADD     (R4)+,R2
 4237  016604  004737  016620          JSR     PC,DECOUT
 4238  016610  005237  016652          INC     R3PBT
 4239  016614  001365                  BNE     TYPT1
```

```
4240  016616  000002                          RTI
4241  016620  005701                DECOUT:   TST     R1
4242  016622  001003                          BNE     DEC1
4243  016624  112720  000060                  MOVB    #60,(R0)+
4244  016630  000207                          RTS     PC
4245  016632  152701  000060        DEC1:     BISB    #60,R1
4246  016636  110120                          MOVB    R1,(R0)+
4247  016640  000207                          RTS     PC
4248
4249  016642  001750                DECPNT:   1000.
4250  016644  000144                          100.
4251  016646  000012                          10.
4252  016650  000001                          1.
4253  016652  000000                R3PBT:    .WORD   0
4254
4255                                ;*************************************************************************
4256                                ;SUBROUTINE TO SET UP 'FK' VECTOR ADDR. & INTR ENABLE
4257                                ;*************************************************************************
4258
4259  016654  017677  000000  162616  LDFKVT:  MOV     @(SP),@FKINT    ;LOAD INTR SERVICE ADDR
4260  016662  062716  000002                  ADD     #2,(SP)
4261  016666  012777  000200  162606          MOV     #200,@FKLVL
4262  016674  052777  000101  162572          BIS     #101,@FKCSR
4263  016702  005077  162404                  CLR     @PSW            ;ENABLE INTERRUPTS
4264  016706  000207                          RTS     PC
4265                                ;*************************************************************************
4266                                ;SUBROUTINE TO RESET 'FK' VECTOR ADDRESSES TO HALT ON INTERRUPTS
4267                                ;*************************************************************************
4268  016710  012777  000340  162374  CLRFKV:  MOV     #340,@PSW
4269  016716  012777  000012  162550          MOV     #12,@FKCSR
4270  016724  013777  001502  162546          MOV     FKLVL,@FKINT
4271  016732  005077  162544                  CLR     @FKLVL
4272  016736  000207                          RTS     PC
4273
4274                                ;*************************************************************************
4275                                ;SUBROUTINE TO LOAD 'VT' VECTOR ADDR. & INTR ENABLE
4276                                ;*************************************************************************
4277
4278  016740  017677  000000  162546  LDVTVT:  MOV     @(SP),@VTINT    ;LOAD INTR SERVICE ADDR.
4279  016746  062716  000002                  ADD     #2,(SP)
4280  016752  012777  000240  162536          MOV     #240,@VTLVL
4281  016760  052777  000101  162522          BIS     #101,@VTCSR     ;SET 'GO' & INTR 'ENABLE'
4282  016766  005077  162320                  CLR     @PSW
4283  016772  000207                          RTS     PC
4284
4285                                ;*************************************************************************
4286                                ;SUBROUTINE TO RESET 'VT' VECTOR ADDRESSES TO HALT ON INTERRUPTS
4287                                ;*************************************************************************
4288
4289  016774  012777  000340  162310  CLRVTV:  MOV     #340,@PSW
4290  017002  042777  000101  162500          BIC     #101,@VTCSR
4291  017010  013777  001516  162476          MOV     VTLVL,@VTINT
4292  017016  005077  162474                  CLR     @VTLVL
4293  017022  000207                          RTS     PC
4294
4295                                ;*************************************************************************
```

```
4296                                    ;SUBROUTINE TO LOAD THE 'FIELD' CONTROL WITH THE '4' MODES AND SET UP
4297                                    ;'VT' BUFFER.
4298                                    ;*************************************************************************
4299
4300  017024  112720  000362          XSTLNE: MOVB    #BLINK,(R0)+            ;LOAD 'BLINK'
4301  017030  104010                          LDLINE                          ;LOAD LINE OF 'BLINKING' CHAR'S
4302  017032  112720  000364                  MOVB    #BOLD,(R0)+
4303  017036  104010                          LDLINE                          ;LOAD LINE OF 'BOLD' CHAR.'S
4304  017040  112720  000361                  MOVB    #BLANK,(R0)+
4305  017044  104010                          LDLINE                          ;LOAD LINE OF 'BLANK' CHAR.'S
4306  017046  112720  000370                  MOVB    #UNLINE,(R0)+
4307  017052  104010                          LDLINE                          ;LOAD LINE OF 'UNDERLINED' CHAR.'S
4308  017054  112720  000360                  MOVB    #CLRFLD,(R0)+
4309  017060  000002                          RTI
4310                                    ;*************************************************************************
4311                                    ;SUBROUTINE TO TEST IF A TTY IS AVAILABLE, AND IF SO. ENABLE
4312                                    ;KEYBOARD INTERRUPTS.
4313                                    ;*************************************************************************
4314
4315  017062  005737  024154          TTYENB: TST     TTYSWH                  ;TTY AVAILABLE?
4316  017066  001405                          BEQ     .+14                    ;NO, EXIT
4317  017070  012777  000100  162216          MOV     #100,@TKS               ;YES, ENABLE TTY INTERRUPTS
4318  017076  005077  162210                  CLR     @PSW
4319  017102  000207                          RTS     PC
4320
4321                                    ;*************************************************************************
4322                                    ;SUBROUTINE TO WAIT FOR AND ASSEMBLE CHARACTERS INPUT
4323                                    ;FROM THE KEYBOARD INTO OCTAL NUMBERS.
4324                                    ;*************************************************************************
4325
4326  017104  104013                  XASEMB: TTYIN                           ;GET CHAR.'S FROM KEYBOARD
4327  017106  012701  014600                  MOV     #INBUF,R1               ;SET UP CHAR. BUFFER POINTER
4328  017112  004737  017200                  JSR     PC,STRIPN               ;STRIPE NO.
4329  017116  010403                  XASEM1: MOV     R4,R3                   ;RETURNS HERE IF ONLY '1' NO.
4330  017120  022703  000701                  CMP     #701,R3
4331  017124  001407                          BEQ     WHY
4332  017126  022703  154765                  CMP     #154765,R3
4333  017132  001401                          BEQ     .+4
4334  017134  000002                          RTI
4335  017136  104012                          PRINT
4336  017140  022704                          MES19
4337  017142  000402                          BR      XASEXT
4338  017144  104012                  WHY:    PRINT
4339  017146  022655                          MES18
4340  017150  000137  002526          XASEXT: JMP     MONITR
4341
4342  017154  005721                  WORD2:  TST     (R1)+                   ;ADVANCE POINTER PAST COMMA
4343  017156  006204                          ASR     R4
4344  017160  006204                          ASR     R4
4345  017162  006204                          ASR     R4
4346  017164  005337  024146                  DEC     CHRCNT                  ;DEC. CHAR. CNTR.
4347  017170  001752                          BEQ     XASEM1                  ;COMMA LAST CHAR.?
4348  017172  010403                          MOV     R4,R3                   ;NO, SAVE 1ST NO.
4349  017174  062716  000002                  ADD     #2,(SP)                 ;SET UP STACK TO EXIT
4350  017200  005004                  STRIPN: CLR     R4
4351  017202  005737  024146                  TST     CHRCNT                  ;WAS ANY DATA INPUTTED?
```

```
4352  017206  001001                        BNE     .+4             ;YES, PROCESS IT
4353  017210  000207                        RTS     PC              ;NO, RETURN
4354  017212  022711  000054                CMP     #54,(R1)        ;CHAR. = COMMA?
4355  017216  001756                        BEQ     WORD2           ;YES, SAVE 1ST NO.
4356  017220  042711  177770                BIC     #177770,(R1)    ;NO, STRIPE NO. TO OCTAL
4357  017224  062104                        ADD     (R1)+,R4
4358  017226  005337  024146                DEC     CHRCNT          ;FINISHED?
4359  017232  003001                        BGT     .+4             ;NO
4360  017234  000207                        RTS     PC              ;YES, EXIT
4361  017236  006304                        ASL     R4
4362  017240  006304                        ASL     R4
4363  017242  006304                        ASL     R4
4364  017244  000756                        BR      STRIPN+2
4365                          ;*************************************************************************
4366                          ;VT MESSAGE ROUTINE, ENTERED VIA EMT DISPATCH HANDLER.
4367                          ;ROUTINE PICKS UP CONTENTS OF THE 'PC' AND USES THIS AS
4368                          ;THE ADDRESS OF MESSAGE TO BE DISPLAYED.
4369                          ;*************************************************************************
4370
4371  017246  004737  017062         VTMES:  JSR     PC,TTYENB       ;ENABLE TTY INTERRUPTS
4372  017252  005077  162232                 CLR     @VTCSR          ;CLR 'GO'
4373  017256  104005                         EOSBUF                  ;PRE-LOAD BUFFER
4374  017260  017637  000000  024466         MOV     @(SP),KSTOR4    ;GET THE MESSAGE ADDRESS FROM STACK
4375  017266  011637  024130                 MOV     (SP),AVECTR     ;GET ADDRESS OF CALL
4376  017272  162737  000002  024130         SUB     #2,AVECTR       ;SAVE AS '+A' RE-START ADDR.
4377  017300  062716  000002                 ADD     #2,(SP)         ;SET UP STACK TO EXIT
4378  017304  104002                         SAVREG                  ;SAVE REG'S
4379  017306  013702  024466                 MOV     KSTOR4,R2
4380  017312  122712  000100         VTERA:  CMPB    #100,(R2)       ;TEST FOR '@'
4381  017316  001007                         BNE     VTER1           ;BRANCH IF NO EQUAL
4382  017320  104003                         GETREG                  ;RESTORE REG'S
4383  017322  005237  024426                 INC     DISPSW          ;SET DISPLAY SW.
4384  017326  104020                         DELAYL
4385  017330  005037  024426                 CLR     DISPSW          ;CLR DISPLAY SW.
4386  017334  000002                         RTI                     ;OTHERWISE EXIT
4387  017336  122712  000045         VTER1:  CMPB    #45,(R2)        ;TEST FOR '%'
4388  017342  001402                         BEQ     VTCRLF          ;IF = TYPE 'CR-LF'
4389  017344  112220                 VTER2:  MOVB    (R2)+,(R0)+     ;SAVE CHAR. IN BUFFER
4390  017346  000761                         BR      VTERA
4391  017350  112720  000012         VTCRLF: MOVB    #12,(R0)+       ;SAVE 'EOL' IN BUFFER
4392  017354  105722                         TSTB    (R2)+           ;INCREMENT BUFFER
4393  017356  000755                         BR      VTERA
4394
4395                          ;*************************************************************************
4396                          ;SUBROUTINE TO TRANSMIT A 'NULL' CHAR. TO THE PRINTER.
4397                          ;*************************************************************************
4398
4399  017360  005037  014326         XNULL:  CLR     SPACEX
4400  017364  105237  014326         1$:     INCB    SPACEX
4401  017370  100375                         BPL     1$
4402  017372  012737  000002  014326         MOV     #2,SPACEX
4403  017400  004737  017436                 JSR     PC,TSTTPS
4404  017404  005077  161712         XNULL1: CLR     @TPB            ;TRANSMIT A NULL CHAR.
4405  017410  004737  017436                 JSR     PC,TSTTPS
4406  017414  005337  014326                 DEC     SPACEX
4407  017420  001371                         BNE     XNULL1
```

```
4408  017422  105237  014326          2$:     INCB    SPACEX
4409  017426  100375                          BPL     2$
4410  017430  005037  014326                  CLR     SPACEX
4411  017434  000002                          RTI
4412
4413  017436  105777  161656          TSTTPS: TSTB    @TPS
4414  017442  100375                          BPL     .-4
4415  017444  000207                          RTS     PC
4416                                  ;*************************************************************************
4417                                  ;ENTERED WITH SYSTEM TRAP 'TRAP' CALL (ERROR) FROM THE 'FK' LOGIC TEST
4418                                  ;THIS ROUTINE SETS UP REG'S 'R0-R5' WITH DATA
4419                                  ;FROM THE 'FK' LOGIC TO BE EXAMINED ON A 'HALT' IF AN 'FK' LOGIC ERROR IS
4420                                  ;DETECTED AND 'NO' TELEPRINTER IS AVAILABLE.
4421                                  ;*************************************************************************
4422
4423  017446  005737  024154          FKERR:  TST     TTYSWH          ;TTY AVAILABLE?
4424  017452  001065                          BNE     TYPFKR          ;YES, TYPE FK ERRORS
4425  017454  052777  000200  162026          BIS     #AUDIO,@VTCSR   ;ISSUE 'AUDIO BEEP' ON ERROR
4426  017462  005777  161636                  TST     @SWR            ;TEST IF 'SW15' IS SET
4427  017466  100014                          BPL     CKHALT          ;NO, INHIBIT ERROR HALT.
4428  017470  013700  024142                  MOV     TSTNUM,R0       ;SAVE 'TEST NO.' IN 'R0'
4429  017474  013701  020404                  MOV     SCOPEF,R1       ;SAVE PASS COUNT
4430  017500  005201                          INC     R1
4431  017502  011602                          MOV     (SP),R2         ;SAVE 'ERROR PC' IN 'R2'
4432  017504  162702  000002                  SUB     #2,R2
4433  017510  017703  161760                  MOV     @FKCSR,R3       ;SAVE 'FK CSR' IN 'R3'
4434  017514  017704  161756                  MOV     @FKDATA,R4      ;SAVE 'FK DATA' IN 'R4'
4435  017520  005777  161600          CKHALT: TST     @SWR            ;CHECK IF SW15 IS SET
4436  017524  100001                          BPL     .+4             ;NO
4437  017526  000000                          HALT                    ;YES, HALT
4438  017530  000002                          RTI
4439
4440                                  ;*************************************************************************
4441                                  ;ENTERED WITH SYSTEM TRAP 'TRAP' CALL (ERROR) FROM THE 'VT' LOGIC TEST
4442                                  ;THIS ROUTINE SETS UP REG'S 'R0-R5' WITH DATA FROM THE 'VT' LOGIC TEST
4443                                  ;TO BE EXAIMNED ON A 'HALT' IF A 'VT' ERROR IS DETECTED AN 'NO' TELEPRINTER
4444                                  ;IS AVAILABLE.
4445                                  ;*************************************************************************
4446
4447  017532  005737  024154          VTERR:  TST     TTYSWH          ;TTY AVAILABLE?
4448  017536  001051                          BNE     TYPVTR          ;YES, TYPE 'VT' ERRORS
4449  017540  005777  161560                  TST     @SWR            ;TEST IF SW15 IS SET
4450  017544  100367                          BPL     CKHALT+4        ;NO, INHIBIT ERROR HALT
4451  017546  104002                          SAVREG                  ;SAVE REGISTERS
4452  017550  010003                          MOV     R0,R3           ;SAVE SHIFT COUNT
4453  017552  010104                          MOV     R1,R4           ;SAVE EXPECTED DATA
4454  017554  013700  024142                  MOV     TSTNUM,R0       ;SAVE FAILING TEST NO.
4455  017560  013701  020404                  MOV     SCOPEF,R1       ;SAVE PASS COUNTER
4456  017564  005201                          INC     R1
4457  017566  011602                          MOV     (SP),R2         :SAVE 'PC'
4458  017570  162702  000002                  SUB     #2,R2
4459  017574  005737  024432                  TST     SHFTSW          ;RUNNING SHIFT TEST'
4460  017600  001005                          BNE     VTERR1          ;YES
4461  017602  017703  161702                  MOV     @VTCSR,R3       ;NO, SAVE CONTENTS OF 'CSR'
4462  017606  017704  161700                  MOV     @VTMAR,R4       ;SAVE CONTENTS OF THE 'MAR'
4463  017612  000402                          BR      VTERR2
```

```
4464  017614  017705  161672          VTERR1: MOV     @VTMAR,R5       ;SAVE CONTENTS OF 'MAR'
4465  017620  000000                  VTERR2: HALT                    ;EXAMINE REG.'S
4466  017622  104003                          GETREG                  ;RESTORE REG.'S
4467  017624  000002                          RTI                     ;EXIT
4468
4469                                  ;*************************************************************************
4470                                  ;ENTERED WITH SYSTEM TRAP 'TRAP' CALL (ERROR) FROM THE 'FK' LOGIC TEST
4471                                  ;THIS ROUTINE IS USED TO PRINT OUT THE CONTENTS OF THE 'FK' REGISTERS
4472                                  ;WHEN A LOGIC ERROR IS DETECTED
4473                                  ;*************************************************************************
4474
4475  017626  012737  022370  020166  TYPFKR: MOV     #MES13,ERRMES   ;SET UP TO PRINT LOGIC ERROR HEADER
4476  017634  012737  000003  024440          MOV     #3,TEMP1        ;SET UP TO PRINT 'FK' REG'S
4477  017642  012701  001474                  MOV     #FKCSR,R1
4478  017646  011637  024464                  MOV     (SP),KSTOR3     ;PC OF FAILING ROUTINE
4479  017652  004737  020120                  JSR     PC,PRTERR       ;PRINT ERROR DATA
4480  017656  000137  017520                  JMP     CKHALT          ;EXIT
4481
4482                                  ;*************************************************************************
4483                                  ;ENTERED WITH SYSTEM TRAP 'TRAP' CALL (ERROR) FROM THE 'VT' LOGIC TEST
4484                                  ;THIS ROUTINE IS USED TO PRINT OUT THE CONTENTS OF THE 'VT' REGISTERS
4485                                  ;WHEN A LOGIC ERROR IS DETECTED
4486                                  ;*************************************************************************
4487
4488  017662  104002                  TYPVTR: SAVREG                  ;SAVE REGISTERS ON STACK
4489  017664  005737  024432                  TST     SHFTSW          ;DOING SHIFT TEST?
4490  017670  001422                          BEQ     VTNORM          ;NO, REPORT AS NORMAL ERROR
4491  017672  005737  024446                  TST     MESPT1          ;TYPED HEADER?
4492  017676  001004                          BNE     .+12            ;YES
4493  017700  005237  024446                  INC     MESPT1
4494  017704  005037  024444                  CLR     MESPRT
4495  017710  012737  022571  020166          MOV     #MES17,ERRMES   ;YES, SET UP SHIFT HEADER
4496  017716  010037  024472                  MOV     R0,KSTOR6       ;SAVE SHIFT COUNT
4497  017722  010137  024470                  MOV     R1,KSTOR5       ;SAVE EXPT'D DATA
4498  017726  012737  000001  024440          MOV     #1,TEMP1        ;PRINT '1' DATA WORD
4499  017734  000413                          BR      VTDUMP
4500  017736  042777  000100  161544  VTNORM: BIC     #100,@VTCSR     ;CLR 'VT' INTR ENABLE
4501  017744  012737  022526  020166          MOV     #MES16,ERRMES   ;SET UP TO PRINT LOGIC ERROR HEADER
4502  017752  012737  000002  024440          MOV     #2,TEMP1        ;SET UP TO PRINT 'VT' REG'S
4503  017760  012701  001510                  MOV     #VTCSR,R1
4504  017764  011637  024464          VTDUMP: MOV     (SP),KSTOR3     ;PC OF FAILING ROUTINE
4505  017770  004737  020120                  JSR     PC,PRTERR       ;PRINT ERROR DATA
4506  017774  032777  020000  161322          BIT     #SW13,@SWR      ;IS PRINT INHIBIT SW. SET?
4507  020002  001043                          BNE     EXTVTR          ;YES, EXIT
4508  020004  005737  024432                  TST     SHFTSW          ;DOING SHIFT TEST?
4509  020010  001414                          BEQ     GETMAR          ;NO, GET AND TYPE 'MAR'
4510  020012  013737  024472  024464          MOV     KSTOR6,KSTOR3   ;YES, SET UP TO TYPE SHIFT COUNT.
4511  020020  004737  020236                  JSR     PC,XPRTA        ;TYPE SHIFT COUNT
4512  020024  104012                          PRINT                   ;PRINT IF DATA IS SHIFTED IN OR OUT
4513  020026  000000          SHFMES: 0
4514  020030  013737  024470  024464          MOV     KSTOR5,KSTOR3   ;SET UP TO TYPE EXPT'D DATA
4515  020036  004737  020236                  JSR     PC,XPRTA        ;TYPE EXPT'D DATA
4516  020042  032777  000007  161254  GETMAR: BIT     #7,@SWR         ;ARE THE 'MAR' SELECT BITS SET?
4517  020050  001412                          BEQ     TYPMAR          ;NO, PRINT MAR AS IS
4518  020052  042777  000017  161430          BIC     #17,@VTCSR      ;YES, RESELECT THE 'MAR'
4519  020060  017702  161240                  MOV     @SWR,R2         ;GET 'MAR' ADDR. FROM SW'S
```

```
4520  020064  042702  177770                    BIC     #177770,R2
4521  020070  006302                            ASL     R2
4522  020072  050277  161412                    BIS     R2,@VTCSR       ;SET UP SELECTED 'MAR' REG.
4523  020076  017737  161410  024464  TYPMAR:   MOV     @VTMAR,KSTOR3
4524  020104  004737  020236                    JSR     PC,XPRTA
4525  020110  104022                            NULL                    ;TRANSMIT 'NULL' CHARACTER
4526  020112  104003                  EXTVTR:   GETREG                  ;RESTORE REGISTERS FROM STACK.
4527  020114  000137  017520                    JMP     CKHALT          ;EXIT
4528
4529                                            ;*************************************************************************
4530                                            ;SUBROUTINE TO PRINT OUT ERROR INFORMATION ON 'FK' & 'VT' LOGIC
4531                                            ;ERRORS.
4532                                            ;*************************************************************************
4533
4534  020120  005737  020462           PRTERR:  TST     XORFLG
4535  020124  001402                            BEQ     .+6
4536  020126  005237  020270                    INC     XORHLT
4537  020132  032777  020000  161164            BIT     #20000,@SWR     ;TEST SW-13 FOR INHIBIT PRINT OUT
4538  020140  001044                            BNE     EXTPRT          ;INHIBIT,CHECK FOR HALT
4539  020142  162737  000002  024464            SUB     #2,KSTOR3
4540  020150  042777  000100  161316            BIC     #100,@FKCSR     ;CLR 'FK' INTR. ENABLE
4541  020156  005737  024444                    TST     MESPRT          ;HEADER BEEN TYPED?
4542  020162  001004                            BNE     .+12            ;YES,
4543  020164  104012                            PRINT
4544  020166  000000                  ERRMES:   0                       ;PRINT LOGIC ERROR HEADER
4545  020170  005237  024444                    INC     MESPRT          ;SET PRINT INHIBIT SW.
4546  020174  104012                            PRINT
4547  020176  023005                            CRLF
4548  020200  104014                            PRTOCT
4549  020202  024142                            TSTNUM                  ;PRINT FAILING TEST NO.
4550  020204  104016                            SPACE
4551  020206  013737  020404  024442            MOV     SCOPEF,TEMP2
4552  020214  005237  024442                    INC     TEMP2
4553  020220  104014                            PRTOCT
4554  020222  024442                            TEMP2                   ;PRINT THE FAILING PASS NO.
4555  020224  104016                            SPACE
4556  020226  000403                            BR      XPRTA
4557  020230  012102                  XPRT:     MOV     (R1)+,R2
4558  020232  011237  024464                    MOV     (R2),KSTOR3
4559  020236  104014                  XPRTA:    PRTOCT
4560  020240  024464                            KSTOR3
4561  020242  104016                            SPACE
4562  020244  005337  024440                    DEC     TEMP1
4563  020250  003367                            BGT     XPRT
4564  020252  005737  020270           EXTPRT:  TST     XORHLT
4565  020256  001403                            BEQ     1$
4566  020260  104012                            PRINT
4567  020262  023752                            MXORH
4568  020264  000000                            HALT
4569
4570  020266  000207                  1$:       RTS     PC
4571  020270  000000                  XORHLT:   000000
4572                                            ;*************************************************************************
4573                                            ;SCOPE AND/OR ITERATION LOOP FOR EACH LOGIC TEST
4574                                            ;*************************************************************************
4575
```

```
4576  020272  104017                    SCOPEC: TSTTKS                  ;CHECK FOR KEYBOARD FLAG
4577  020274  032777  040000  161022            BIT     #40000,@SWR     ;TEST SW-14 FOR SCOPE
4578  020302  001015                            BNE     SCOPEB          ;YES, SCOPE
4579  020304  005737  020462                    TST     XORFLG          ;SEE IF RUNNING WITH XOR
4580  020310  001037                            BNE     XORT
4581  020312  032777  004000  161004  SCOP1:    BIT     #4000,@SWR      ;NO-TEST SW-11 FOR ITERATION
4582  020320  001013                            BNE     SCOPEG          ;INHIBIT ITERATION
4583  020322  023737  020404  020402            CMP     SCOPEF,ICOUNT   ;COMPARE CURRENT COUNT TO MAX NUMBER
4584  020330  100007                            BPL     SCOPEG          ;EXIT-DONE
4585  020332  005237  020404                    INC     SCOPEF          ;INCREMENT COUNT
4586  020336  022606                SCOPEB:     CMP     (6)+,SP         ;REPOSITION STACK
4587  020340  012677  160746                    MOV     (6)+,@PSW       ;RESTORE PREVIOUS PROCESSOR STATUS
4588  020344  000177  000036                    JMP     @RETURN         ;REPEAT TEST
4589  020350  005037  020404          SCOPEG:   CLR     SCOPEF          ;CLEAR COUNT
4590  020354  011601                            MOV     @SP,R1          ;SAVE TEST NO.
4591  020356  011137  024142                    MOV     (R1),TSTNUM
4592  020362  062716  000002                    ADD     #2,(SP)
4593  020366  011637  020406                    MOV     @SP,RETURN      ;SAVE SCOPE RETURN POINTER
4594  020372  052777  000200  161110            BIS     #AUDIO,@VTCSR   ;ISSUE 'AUDIO BEEP' ON END OF TEST
4595  020400  000002                            RTI                     ;RETURN INLINE-NEXT TEST
4596  020402  000200                ICOUNT:     200                     ;ITERATION COUNT
4597  020404  000000                SCOPEF:     0                       ;COUNT LOCATION FOR ITERATION LOOP
4598  020406  006246                RETURN:     FKT1
4599                                ;*****BEGINNING OF XOR TESTING
4600  020410  013746  000004          XORT:     MOV     @#4,    -(6)    ;SAVE CONTENTS OF LOCATION 4
4601  020414  012737  020434  000004            MOV     #XORERR,@#4     ;SET UP FOR XOR TIMEOUT
4602  020422  005737  177060                    TST     @#177060        ;IF XOR TIMES OUT IT DETECTED AN ERROR
4603  020426  012637  000004                    MOV     (6)+,@#4        ;NO TIMEOUT-RESTORE LOCATION 4
4604  020432  000727                            BR      SCOP1           ;RETURN TO TEST
4605  020434  022626                XORERR:     CMP     (6)+,(6)+       ;CLEAR STACK
4606  020436  012637  000004                    MOV     (6)+,@#4        ;RESTORE LOCATION 4
4607  020442  032777  000001  160654            BIT     #1,     @SWR    ;RING BELL ON ERROR?
4608  020450  001402                            BEQ     1$              ;IF NO, SKIP AHEAD
4609  020452  104012                            PRINT                   ;SEND BELL
4610  020454  023750                            MBELL
4611  020456  000177  177724          1$:       JMP     @RETURN
4612  020462  000000                XORFLG:     000000
4613
4614
4615                                ;*************************************************************************
4616                                ;KEYBOARD CHARACTER TABLE
4617                                ;*************************************************************************
4618                                                                    ;START, TOP UN-SHIFTED
4619  020464      133               CHRTAB:     .BYTE   '[
4620  020465      061                           .BYTE   '1
4621  020466      062                           .BYTE   '2
4622  020467      063                           .BYTE   '3
4623  020470      064                           .BYTE   '4
4624  020471      065                           .BYTE   '5
4625  020472      066                           .BYTE   '6
4626  020473      067                           .BYTE   '7
4627  020474      070                           .BYTE   '8
4628  020475      071                           .BYTE   '9
4629  020476      060                           .BYTE   '0
4630  020477      055                           .BYTE   '-
4631  020500      137                           .BYTE   '←
```

```
 4632  020501  135                     .BYTE   ']
 4633
 4634                                  ;2ND ROW LOWER CASE
 4635
 4636  020502  011                     .BYTE   11              ;TAB
 4637  020503  033                     .BYTE   33              ;ALTMODE
 4638  020504  161                     .BYTE   161             ;Q
 4639  020505  167                     .BYTE   167             ;W
 4640  020506  145                     .BYTE   145             ;E
 4641  020507  162                     .BYTE   162             ;R
 4642  020510  164                     .BYTE   164             ;T
 4643  020511  171                     .BYTE   171             ;Y
 4644  020512  165                     .BYTE   165             ;U
 4645  020513  151                     .BYTE   151             ;I
 4646  020514  157                     .BYTE   157             ;O
 4647  020515  160                     .BYTE   160             ;P
 4648  020516  100                     .BYTE   '@
 4649  020517  012                     .BYTE   12              ;'LF'
 4650  020520  015                     .BYTE   15              ;'CR'
 4651
 4652                                  ;3RD ROW LOWER CASE
 4653
 4654  020521  136                     .BYTE   '^
 4655  020522  141                     .BYTE   141             ;A
 4656  020523  163                     .BYTE   163             ;S
 4657  020524  144                     .BYTE   144             ;D
 4658  020525  146                     .BYTE   146             ;F
 4659  020526  147                     .BYTE   147             ;G
 4660  020527  150                     .BYTE   150             ;H
 4661  020530  152                     .BYTE   152             ;J
 4662  020531  153                     .BYTE   153             ;K
 4663  020532  154                     .BYTE   154             ;L
 4664  020533  073                     .BYTE   ';
 4665  020534  072                     .BYTE   ':
 4666  020535  177                     .BYTE   177             ;RUBOUT
 4667
 4668                                  ;4TH ROW LOWER CASE
 4669
 4670  020536  172                     .BYTE   172             ;Z
 4671  020537  170                     .BYTE   170             ;X
 4672  020540  143                     .BYTE   143             ;C
 4673  020541  166                     .BYTE   166             ;V
 4674  020542  142                     .BYTE   142             ;B
 4675  020543  156                     .BYTE   156             ;N
 4676  020544  155                     .BYTE   155             ;M
 4677  020545  054                     .BYTE   ',
 4678  020546  056                     .BYTE   '.
 4679  020547  057                     .BYTE   '/
 4680  020550  134                     .BYTE   '\
 4681  020551  040                     .BYTE   40                      ;TERMINATE W. SPACE
 4682
 4683                                  ;START OF 'SHIFTED' PASS
 4684
 4685  020552  173             JPCASE: .BYTE   173             ;LEFT BRACE
 4686  020553  041                     .BYTE   '!
 4687  020554  042                     .BYTE   '"
```

```
 4688  020555  043                    .BYTE   '#
 4689  020556  044                    .BYTE   '$
 4690  020557  045                    .BYTE   '%
 4691  020560  046                    .BYTE   '&
 4692  020561  047                    .BYTE   '
 4693  020562  050                    .BYTE   '(
 4694  020563  051                    .BYTE   ')
 4695  020564  060                    .BYTE   '0
 4696  020565  075                    .BYTE   '=
 4697  020566  137                    .BYTE   '←
 4698  020567  175                    .BYTE   175             ;RIGHT BRACE
 4699
 4700                          ;2ND ROW 'SHIFTED'
 4701
 4702  020570  011                    .BYTE   11
 4703  020571  033                    .BYTE   33              ;ALTMODE
 4704  020572  121                    .BYTE   'Q
 4705  020573  127                    .BYTE   'W
 4706  020574  105                    .BYTE   'E
 4707  020575  122                    .BYTE   'R
 4708  020576  124                    .BYTE   'T
 4709  020577  131                    .BYTE   'Y
 4710  020600  125                    .BYTE   'U
 4711  020601  111                    .BYTE   'I
 4712  020602  117                    .BYTE   'O
 4713  020603  120                    .BYTE   'P
 4714  020604  140                    .BYTE   140
 4715  020605  012                    .BYTE   12              ;'LF.
 4716  020606  015                    .BYTE   15              ;'CR'
 4717
 4718                          ;3RD ROW 'SHIFTED'
 4719
 4720  020607  176                    .BYTE   176             ;'↑'
 4721  020610  101                    .BYTE   'A
 4722  020611  123                    .BYTE   'S
 4723  020612  104                    .BYTE   'D
 4724  020613  106                    .BYTE   'F
 4725  020614  107                    .BYTE   'G
 4726  020615  110                    .BYTE   'H
 4727  020616  112                    .BYTE   'J
 4728  020617  113                    .BYTE   'K
 4729  020620  114                    .BYTE   'L
 4730  020621  053                    .BYTE   '+
 4731  020622  052                    .BYTE   '*
 4732  020623  177                    .BYTE   177             ;RLBOUT
 4733
 4734                          ;4TH ROW 'SHIFTED'
 4735
 4736  020624  132                    .BYTE   'Z
 4737  020625  130                    .BYTE   'X
 4738  020626  103                    .BYTE   'C
 4739  020627  126                    .BYTE   'V
 4740  020630  102                    .BYTE   'B
 4741  020631  116                    .BYTE   'N
 4742  020632  115                    .BYTE   'M
 4743  020633  074                    .BYTE   '<
```

```
4744  020634  076                     .BYTE   '>
4745  020635  077                     .BYTE   '?
4746  020636  174                     .BYTE   174
4747  020637  040                     .BYTE   40                      ;TERMINATE W/ SPACE
4748
4749                          ;START OF 'CONTROL' PASS
4750
4751  020640  033             CONTRL: .BYTE   33                      ;'↑['
4752  020641  061                     .BYTE   '1
4753  020642  062                     .BYTE   '2
4754  020643  063                     .BYTE   '3
4755  020644  064                     .BYTE   '4
4756  020645  065                     .BYTE   '5
4757  020646  066                     .BYTE   '6
4758  020647  067                     .BYTE   '7
4759  020650  070                     .BYTE   '8
4760  020651  071                     .BYTE   '9
4761  020652  060                     .BYTE   '0
4762  020653  015             CNTRIT: .BYTE   15
4763  020654  037                     .BYTE   37                      ;'↑←'
4764  020655  035                     .BYTE   35                      ;'↑'RIGHT BRACKET
4765
4766                          ;2ND ROW 'CONTROL'
4767
4768  020656  011                     .BYTE   11
4769  020657  033                     .BYTE   33                      ;ALTMODE
4770  020660  021                     .BYTE   21                      ;'↑Q'
4771  020661  027                     .BYTE   27                      ;'↑W'
4772  020662  005                     .BYTE   05                      ;'↑E'
4773  020663  022                     .BYTE   22                      ;'↑R'
4774  020664  024                     .BYTE   24                      ;'↑T'
4775  020665  031                     .BYTE   31                      ;'↑Y'
4776  020666  025                     .BYTE   25                      ;'↑U'
4777  020667  011                     .BYTE   11                      ;'↑I'
4778  020670  017                     .BYTE   17                      ;'↑O'
4779  020671  020                     .BYTE   20                      ;'↑P'
4780  020672  000                     .BYTE   00                      ;'↑@'
4781  020673  012                     .BYTE   12                      ;'LF'
4782  020674  015                     .BYTE   15                      ;'CR'
4783
4784                          ;3RD ROW 'CONTROL'
4785
4786  020675  036                     .BYTE   36                      ;'↑↑'
4787  020676  001                     .BYTE   01                      ;'↑A'
4788  020677  023                     .BYTE   23                      ;'↑S'
4789  020700  004                     .BYTE   04                      ;'↑D'
4790  020701  006                     .BYTE   06                      ;'↑F'
4791  020702  007                     .BYTE   07                      ;'↑G'
4792  020703  010                     .BYTE   10                      ;'↑H'
4793  020704  012                     .BYTE   12                      ;'↑J'
4794  020705  013                     .BYTE   13                      ;'↑K'
4795  020706  014                     .BYTE   14                      ;'↑L'
4796  020707  073                     .BYTE   73                      ;';'
4797  020710  072                     .BYTE   72                      ;':'
4798  020711  177                     .BYTE   177                     ;'↑'RUBOJT
4799
```

```
                                        ;4TH ROW 'CONTROL'

                        .BYTE   32              ;'↑Z'
                        .BYTE   30              ;'↑X'
                        .BYTE   03              ;'↑C'
                        .BYTE   26              ;'↑V'
                        .BYTE   02              ;'↑B'
                        .BYTE   16              ;'↑N'
                        .BYTE   15              ;'↑M'
                        .BYTE   ',
                        .BYTE   '.
                        .BYTE   '/
                        .BYTE   34
CHREND:                 .BYTE   40              ;SPACE,  'DONE'
                        .EVEN
                        ;*************************************************************************
                        ;'VT' SHIFT DATA TABLE USED BY THE 'VT' LOGIC SHIFT TEST
                        ;THIS BUFFER IS SHIFTED IN 2 PASSES (64 TIMES IN ALL) TO TEST THE
                        ;SHIFT REGISTER INPUT & OUTPUT LOGIC. AFTER EACH NPR TRANSFER IS MADE
                        ;THE DATA FROM THIS TABLE IS COMPARED TO THE CONTENTS OF THE MAINTENANCE
                        ;ADDRESS REGISTER. THE SHIFTS ARE COMMENTED IN OCTAL AND REPORTED IN
                        ;OCTAL. THE LOW BYTE OF EACH SHIFT IS A CONTROL FIELD CHARACTER AND
                        ;THE HIGH BYTE IS VALID DATA CHARACTER.
                        ;*************************************************************************

SHFTBF:                 .BYTE   360
                        .BYTE   40              ;SHIFT '1' OR '41'
                        .BYTE   361
                        .BYTE   41              ;SHIFT '2' OR '42'
                        .BYTE   362
                        .BYTE   42              ;SHIFT '3' OR '43'
                        .BYTE   364
                        .BYTE   43              ;SHIFT '4' OR '44'
                        .BYTE   370
                        .BYTE   44              ;SHIFT '5' OR '45'
                        .BYTE   363
                        .BYTE   45              ;SHIFT '6' OR '46'
                        .BYTE   365
                        .BYTE   46              ;SHIFT '7' OR '47'
                        .BYTE   367
                        .BYTE   47              ;SHIFT '10' OR '50'
                        .BYTE   371
                        .BYTE   50              ;SHIFT '11' OR '51'
                        .BYTE   373
                        .BYTE   61              ;SHIFT '12' OF '52'
                        .BYTE   375
                        .BYTE   72              ;SHIFT '13' OR '53'
                        .BYTE   377
                        .BYTE   107             ;SHIFT '14' OR '54'
                        .BYTE   366
                        .BYTE   106             ;SHIFT '15' OR '55'
                        .BYTE   372
                        .BYTE   170             ;SHIFT '16' OR '56'
                        .BYTE   374
                        .BYTE   123             ;SHIFT '17' OR '57'
                        .BYTE   376
```

```
4856  020765  137                       .BYTE   137             ;SHIFT '20' OR '60'
4857  020766  360                       .BYTE   360
4858  020767  177                       .BYTE   177             ;SHIFT '21' OR '61'
4859  020770  360                       .BYTE   360
4860  020771  200                       .BYTE   200             ;SHIFT '22' OR '62'
4861  020772  376                       .BYTE   376
4862  020773  237                       .BYTE   237             ;SHIFT '23' OR '63'
4863  020774  374                       .BYTE   374
4864  020775  277                       .BYTE   277             ;SHIFT '24' OR '64'
4865  020776  372                       .BYTE   372
4866  020777  253                       .BYTE   253             ;SHIFT '25' OR '65'
4867  021000  366                       .BYTE   366
4868  021001  235                       .BYTE   235             ;SHIFT '26' OR '66'
4869  021002  360                       .BYTE   360
4870  021003  040                       .BYTE   40              ;SHIFT '27' OR '67'
4871  021004  361                       .BYTE   361
4872  021005  041                       .BYTE   41              ;SHIFT '30' OR '70'
4873  021006  363                       .BYTE   363
4874  021007  043                       .BYTE   43              ;SHIFT '31' OR '71'
4875  021010  362                       .BYTE   362
4876  021011  045                       .BYTE   45              ;SHIFT '32' OR '72'
4877  021012  364                       .BYTE   364
4878  021013  047                       .BYTE   47              ;SHIFT '33' OR '73'
4879  021014  366                       .BYTE   366
4880  021015  050                       .BYTE   50              ;SHIFT '34' OR '74'
4881  021016  365                       .BYTE   365
4882  021017  070                       .BYTE   70              ;SHIFT '35' OR '75'
4883  021020  367                       .BYTE   367
4884  021021  071                       .BYTE   71              ;SHIFT '36' OR '76'
4885  021022  370                       .BYTE   370
4886  021023  072                       .BYTE   72              ;SHIFT '37' OR '77'
4887  021024  360               SHFEND: .BYTE   360
4888  021025  031                       .BYTE   31              ;SHIFT '40' OR '100'
4889
4890                            ;*****************************************************************************
4891                            ;MESSAGES
4892                            ;*****************************************************************************
4893  021026  000                       .BYTE
4894  021027  045  053045  031124 TITLE:  .ASCII  '%%VT20 DIAGNOSTIC TEST MAINDEC-11-DBVTA-D-PB @'
4895  021034  020060  044504  043501
4896  021042  047516  052123  041511
4897  021050  052040  051505  020124
4898  021056  040515  047111  042504
4899  021064  026503  030461  042055
4900  021072  053102  040524  042055
4901  021100  050055  020102    100
4902
4903  021105  042  045506  051440 MES1:   .ASCII  '"FK SWITCH & LIGHT TEST", READY FOR INPUT%@'
4904  021112  044527  041524  020110
4905  021120  020046  044514  044107
4906  021126  020124  042524  052123
4907  021134  026042  051040  040505
4908  021142  054504  043040  051117
4909  021150  044440  050116  052125
4910  021156  040045
4911  021160  040445  042522  042040 MES1A:  .ASCII :%ARE DEFAULT ADDRESSES & VECTORS OK (Y OR N) ^ @:
```

```
4912  021166  043105  052501  052114
4913  021174  040440  042104  042522
4914  021202  051523  051505  023040
4915  021210  053040  041505  047524
4916  021216  051522  047440  020113
4917  021224  054450  047440  020122
4918  021232  024516  037440  040040
4919
4920  021240  021045  045506  046040  MES2:   .ASCII  '%"FK LOGIC TEST"%%@'
4921  021246  043517  041511  052040
4922  021254  051505  021124  022445
4923  021262     100
4924  021263     045  042524  052123  MES3:   .ASCII  ;%TEST NO. (0-22) ....%@;
4925  021270  047040  027117  024040
4926  021276  026460  031062  020051
4927  021304  027056  027056  040045
4928
4929  021312  043042  020113  044103  MES4:   .ASCII  ;"FK CHARACTER TEST",INPUT 'LOWER CASE' CHARACTERS.%@;
4930  021320  051101  041501  042524
4931  021326  020122  042524  052123
4932  021334  026042  047111  052520
4933  021342  020124  046047  053517
4934  021350  051105  041440  051501
4935  021356  023505  041440  040510
4936  021364  040522  052103  051105
4937  021372  027123  040045
4938
4939  021376  052045  041125  020105  MES5:   .ASCII  ;%TUBE '0', '1' ,'2','3', OR '4'? @;
4940  021404  030047  026047  023440
4941  021412  023461  026040  031047
4942  021420  026047  031447  026047
4943  021426  047440  020122  032047
4944  021434  037447  040040
4945
4946  021440  043045  051117  042511  MFCP:   .ASCII  /%FORIEGN CHARACTER SET ('Y' OR 'N' ?)@/
4947  021446  047107  041440  040510
4948  021454  040522  052103  051105
4949  021462  051440  052105  024040
4950  021470  054447  020047  051117
4951  021476  023440  023516  037440
4952  021504  040051
4953  021506  044445  050116  052125  MES6:   .ASCII  ;%INPUT CHAR., TEST 1.@;
4954  021514  041440  040510  027122
4955  021522  020054  042524  052123
4956  021530  030447  040056
4957  021534  044445  050116  052125  MES6A:  .ASCII  ;%INPUT CHAR., TEST 2.@;
4958  021542  041440  040510  027122
4959  021550  020054  042524  052123
4960  021556  031040  040056
4961  021562  044445  050116  052125  MES6B:  .ASCII  ;%INPUT CHAR., TEST 3.@;
4962  021570  041440  040510  027122
4963  021576  020054  042524  052123
4964  021604  031440  040056
4965
4966  021610  021045  042524  052123  MES7:   .ASCII  ;%"TEST COMPLETE"   @;
4967  021616  041440  045517  046120
```

```
 4968  021624  052105  021105  020040
 4969  021632  040040
 4970
 4971
 4972  021634  042524  052123  040440  MES8:   .ASCII  ;TEST ADDR.? @;
 4973  021642  042104  027122  020077
 4974  021650     100
 4975
 4976  021651     042  052126  030062  MES9:   .ASCII  ;"VT20 SYSTEM EXERCISER TEST". ;
 4977  021656  051440  051531  042524
 4978  021664  020115  054105  051105
 4979  021672  044503  042523  020122
 4980  021700  042524  052123  027042
 4981  021706     040
 4982  021707     012                          .BYTE   12
 4983  021710     012                          .BYTE   12
 4984  021711     116  052117  035105  .ASCII  ;NOTE: THE HOST DIAGNOSTIC, EITHER FOR THE PDP-8 OR 11 MUST BE;
 4985  021716  052040  042510  044040
 4986  021724  051517  020124  044504
 4987  021732  043501  047516  052123
 4988  021740  041511  020054  044505
 4989  021746  044124  051105  043040
 4990  021754  051117  052040  042510
 4991  021762  050040  050104  034055
 4992  021770  047440  020122  030461
 4993  021776  046440  051525  020124
 4994  022004  042502
 4995  022006     012                          .BYTE   12
 4996  022007     114  040517  042504          .ASCII  ;LOADED INTO THE HOST COMPUTER BEFORE RUNNING THIS TEST.;
 4997  022014  020104  047111  047524
 4998  022022  052040  042510  044040
 4999  022030  051517  020124  047503
 5000  022036  050115  052125  051105
 5001  022044  041040  043105  051117
 5002  022052  020105  052522  047116
 5003  022060  047111  020107  044124
 5004  022066  051511  052040  051505
 5005  022074  027124
 5006  022076     031                          .BYTE   31
 5007
 5008  022077     042  052126  047440  MES10:  .ASCII  ;"VT ODD ADDRESS TEST".@;
 5009  022104  042104  040440  042104
 5010  022112  042522  051523  052040
 5011  022120  051505  021124  040056
 5012
 5013
 5014  022126  044445  046114  043505  MESSA:  .ASCII  ;%ILLEGAL TRAP TO @;
 5015  022134  046101  052040  040522
 5016  022142  020120  047524  040040
 5017
 5018  022150  043040  047522  020115  MES6C:  .ASCII  ; FROM @;
 5019  022156     100
 5020  022157     103  040510  051522  MES80:  .ASCII  ;CHARS.: 0000  RECV'D: 0000  XFERS.: 0000  ;
 5021  022164  035056  030040  030060
 5022  022172  020060  051040  041505
 5023  022200  023526  035104  030040
```

```
5024  022206  030060  020060  054040
5025  022214  042506  051522  035056
5026  022222  030040  030060  020060
5027  022230     040
5028  022231     105  051122  027123          .ASCII  ;ERRS.: 0000;
5029  022236  020072  030060  030060
5030  022244     012                          .BYTE   12
5031  022245     012                          .BYTE   12
5032  022246     031                          .BYTE   31
5033
5034          022250                          .EVEN
5035  022250     000                          .BYTE
5036  022251     124  044510  020123  MES11:  .ASCII  ;THIS MESSAGE IS BEING LOADED FROM AN ODD STARTING ;
5037  022256  042515  051523  043501
5038  022264  020105  051511  041040
5039  022272  044505  043516  046040
5040  022300  040517  042504  020104
5041  022306  051106  046517  040440
5042  022314  020116  042117  020104
5043  022322  052123  051101  044524
5044  022330  043516     040
5045  022333     101  042104  042522          .ASCII  ;ADDRESS. HOW  DOES IT LOOK??;
5046  022340  051523  020056  047510
5047  022346  020127  042040  042517
5048  022354  020123  052111  046040
5049  022362  047517  037513     077
5050  022367     031                          .BYTE   31
5051  022370  052045  052123  047040  MES13:  .ASCII  ;%TST NO  PASS    MA    CSR    DATA@;
5052  022376  020117  050040  051501
5053  022404  020123  020040  046440
5054  022412  020101  020040  041440
5055  022420  051123  020040  020040
5056  022426  040504  040524     100
5057  022433     042  045506  051040  MES14:  .ASCII  ;"FK REPEATIBILITY TEST", READY FOR INPUT.%@;
5058  022440  050105  040505  044524
5059  022446  044502  044514  054524
5060  022454  052040  051505  021124
5061  022462  020054  042522  042101
5062  022470  020131  047506  020122
5063  022476  047111  052520  027124
5064  022504  040045
5065  022506  041445  040510  027122  MES15:  .ASCII  ;%CHAR. CODE(S) @;
5066  022514  041440  042117  024105
5067  022522  024523  040040
5068
5069  022526  022445  051524  020124  MES16:  .ASCII  ;%%TST NO  PASS    MA    CSR    MAR@;
5070  022534  047516  020040  040520
5071  022542  051523  020040  020040
5072  022550  040515  020040  020040
5073  022556  051503  020122  020040
5074  022564  046440  051101     100
5075
5076  022571     045  052045  052123  MES17:  .ASCII  ;%%TST NO  PASS    MA   SHIFT  IN/OUT  EXPT'D RECV'D@;
5077  022576  047040  020117  050040
5078  022604  051501  020123  020040
5079  022612  046440  020101  020040
```

```
5080  022620  044123  043111  020124
5081  022626  044440  027516  052517
5082  022634  020124  042440  050130
5083  022642  023524  020104  042522
5084  022650  053103  042047     100
5085  022655     045  042502  040503  MES18:  .ASCII  ;%BECAUSE I LOVE YOU!!%@;
5086  022662  051525  020105  020111
5087  022670  047514  042526  054440
5088  022676  052517  020441  040045
5089  022704  044445  020106  047516  MES19:  .ASCII  ;%IF NOT WITH THE ONE YOU LOVE,;
5090  022712  020124  044527  044124
5091  022720  052040  042510  047440
5092  022726  042516  054440  052517
5093  022734  046040  053117  026105
5094  022742  046040  053117  020105          .ASCII  ; LOVE THE ONE YOUR WITH.%@;
5095  022750  044124  020105  047117
5096  022756  020105  047531  051125
5097  022764  053440  052111  027110
5098  022772  040045
5099
5100  022774  041536  027045     100  CNTRLC: .ASCII  '↑C%.@'
5101  023001     136  022501     100  CNTRLA: .ASCII  '↑A%@'
5102
5103  023005     045     100          CRLF:   .ASCII  '%@'
5104
5105
5106  023007     077  020040     100  QMARK:  .ASCII  '?  @'
5107
5108
5109  023013     045  053042  031124  MES20:  .ASCII  ;%"VT20 LOGIC TEST"%@;
5110  023020  020060  047514  044507
5111  023026  020103  042524  052123
5112  023034  022442     100
5113
5114  023037     045  051045  041505  MES21:  .ASCII  '%%RECOVERED FROM POWER FAILURE @'
5115  023044  053117  051105  042105
5116  023052  043040  047522  020115
5117  023060  047520  042527  020122
5118  023066  040506  046111  051125
5119  023074  020105     100
5120
5121  023077     111  050116  052125  MES22:  .ASCII  ;INPUT THE 'UPPER CASE' CHARACTERS.%@;
5122  023104  052040  042510  023440
5123  023112  050125  042520  020122
5124  023120  040503  042523  020047
5125  023126  044103  051101  041501
5126  023134  042524  051522  022456
5127  023142     100
5128
5129  023143     111  050116  052125  MES23:  .ASCII  ;INPUT THE 'CONTROL' CHARACTERS.%@;
5130  023150  052040  042510  023440
5131  023156  047503  052116  047522
5132  023164  023514  041440  040510
5133  023172  040522  052103  051105
5134  023200  027123  040045
5135  023204  053042  020124  044103  MES24:  .ASCII                          ;"VT CHARACTER DISPLAY TEST"@;
```

```
5136  023212  051101  041501  042524
5137  023220  020122  044504  050123
5138  023226  040514  020131  042524
5139  023234  052123  040042
5140  023240  053042  020124  044506  MES25:  .ASCII                          ;"VT FIELD CONTROL TEST"@;
5141  023246  046105  020104  047503
5142  023254  052116  047522  020114
5143  023262  042524  052123  040042
5144  023270  053042  020124  051503  MES26:  .ASCII                          ;"VT CSR PRESET TEST"@;
5145  023276  020122  051120  051505
5146  023304  052105  052040  051505
5147  023312  021124     100
5148  023315     042  052126  046040  MES27:  .ASCII                          ;"VT LINE FEED TEST"@;
5149  023322  047111  020105  042506
5150  023330  042105  052040  051505
5151  023336  021124     100
5152  023341     042  052126  042440  MES30:  .ASCII                          ;"VT END OF SCREEN TEST"@;
5153  023346  042116  047440  020106
5154  023354  041523  042522  047105
5155  023362  052040  051505  021124
5156  023370     100
5157  023371     042  052126  041040  MES31:  .ASCII                          ;"VT BLANK CONTROL TEST"@;
5158  023376  040514  045516  041440
5159  023404  047117  051124  046117
5160  023412  052040  051505  021124
5161  023420     100
5162  023421     042  052126  040440  MES33:  .ASCII                          ;"VT ALIGNMENT TEST"@;
5163  023426  044514  047107  042515
5164  023434  052116  052040  051505
5165  023442  021124     100
5166  023445     042  052126  043040  MES34:  .ASCII                          ;"VT FOCUS TEST"@;
5167  023452  041517  051525  052040
5168  023460  051505  021124     100
5169  023465     042  052126  053440  MES35:  .ASCII                          ;"VT WORST CASE TEST"@;
5170  023472  051117  052123  041440
5171  023500  051501  020105  042524
5172  023506  052123  040042
5173  023512  053042  020124  052503  MES36:  .ASCII                          ;"VT CURSOR MOVEMENT TEST"@;
5174  023520  051522  051117  046440
5175  023526  053117  046505  047105
5176  023534  020124  042524  052123
5177  023542  040042
5178
5179  023544  020040  047111  020040  MES37:  .ASCII          :  IN    @:
5180  023552  020040     100
5181
5182  023555     040  052517  020124  MES38:  .ASCII          : OUT    @:
5183  023562  020040  040040
5184
5185  023566  041042  046117  020104  MES39:  .ASCII  ;"BOLD PRESET TEST"@;
5186  023574  051120  051505  052105
5187  023602  052040  051505  021124
5188  023610     100
5189
5190  023611     042  046102  047111  MES40:  .ASCII  ;"BLINK PRESET TEST"@;
5191  023616  020113  051120  051505
```

```
5192  023624  052105  052040  051505
5193  023632  021124     100
5194
5195  023635     042  046102  047101  MES41:  .ASCII  ;"BLANK PRESET TEST"@;
5196  023642  020113  051120  051505
5197  023650  052105  052040  051505
5198  023656  021124     100
5199
5200  023661     042  047125  042504  MES42:  .ASCII  ;"UNDERLINE PRESET TEST"@;
5201  023666  046122  047111  020105
5202  023674  051120  051505  052105
5203  023702  052040  051505  021124
5204  023710     100
5205  023711     045  052522  047116  MES43:  .ASCII  ;%RUNNING TUBE 0@;
5206  023716  047111  020107  052524
5207  023724  042502  030040     100
5208
5209  023731     045  052522  047116  MES44:  .ASCII  ;%RUNNING TUBE @;
5210  023736  047111  020107  052524
5211  023744  042502  040040
5212
5213  023750  040007                  MBELL:  .ASCII  <7>/@/
5214  023752  050045  047522  051107  MXORH:  .ASCII  /%PROGRAM RUN IS ABORTED@/
5215  023760  046501  051040  047125
5216  023766  044440  020123  041101
5217  023774  051117  042524  040104
5218  024002  047045  046525  042502  MED1:   .ASCII  /%NUMBER OF TUBES? @/
5219  024010  020122  043117  052040
5220  024016  041125  051505  020077
5221  024024     100
5222          024026                  .EVEN
5223  024026  053045  020124     100  MUTP:   .ASCII  /%VT @/
5224  024033     045  045506     040  MFKP:   .ASCII  /%FK /
5225  024037     040  042526  052103  MAUFK:  .ASCII  / VECTOR AND ADDRESS FOR TUBE @/
5226  024044  051117  040440  042116
5227  024052  040440  042104  042522
5228  024060  051523  043040  051117
5229  024066  052040  041125  020105
5230  024074     100
5231  024075     045  042040  030514  MDLVA:  .ASCII  /% DL11 VECTOR FOR TUBE @/
5232  024102  020061  042526  052103
5233  024110  051117  043040  051117
5234  024116  052040  041125  020105
5235  024124     100
5236          024126                  .EVEN
5237                                  ;ADDRESS AND CONSTANTS TABLE
5238
5239  024126  000000                  PIMP:   0
5240  024130  001726                  AVECTR: INITB                   ;CNTR 'A' VECTOR ADDRESS
5241  024132  001000                  STACK:  1000                    ;'SP' INITIALIZATION ADDRESS
5242  024134  000000                  UNITFG: 0                       ;SOFTWARE SW., SET IF BOTH UNITS AVAIL.
5243  024136  000000                  UNITNO: 0                       ;CONTAINS NO. OF UNITS AVAILABLE
5244  024140  000000                  MEMSIZ: 0                       ;CONTAINS HIGHEST 'K' OF MEM. ADDR.
5245  024142  000000                  TSTNUM: 0                       ;CONTAINS CURRENT LOGIC TEST NO.
5246  024144  000000                  FIELDSW:0                       ;FIELD CONTROL SOFTWARE SW.
5247  024146  000000                  CHRCNT: 0                       ;'VT' BUFFER CHARACTER COUNTER
```

```
5248  024150  000000          RUNSWH: 0                       ;'RUN ALL' SOFTWARE SW.
5249  024152  000000          INTSWH: 0                       ;SOFTWARE SW.
5250  024154  000000          TTYSWH: 0                       ;SOFTWARE SW., SET IF TTY AVAIBLE
5251  024156  000000          TOPC:   0
5252  024160  000000          FROMPC: 0
5253  024162  000000          DONESW: 0
5254  024164  000000          SWAPEM: 0
5255  024166  000000          MTRSWH: 0                       ;SOFTWARE SW.
5256  024170  000000          RECSW0: 0                       ;SET IF RECEIVING OR EXPECTING DATA
5257  024172  000000          RECSW1: 0
5258  024174  000000          RECSW2: 0
5259  024176  000000          RECSW3: 0
5260  024200  000000          TRNSW0: 0                       ;SET IF TRANSMITTING DATA
5261  024202  000000          TRNSW1: 0
5262  024204  000000          TRNSW2: 0
5263  024206  000000          TRNSW3: 0
5264  024210  000000          NULCT0: 0
5265  024212  000000          NULCT1: 0
5266  024214  000000          NULCT2: 0
5267  024216  000000          NULCT3: 0
5268  024220  000000          STCOD0: 0                       ;SET IF START CODE (377) WAS RECV'D
5269  024222  000000          STCOD1: 0
5270  024224  000000          STCOD2: 0
5271  024226  000000          STCOD3: 0
5272  024230  000000          CHRCT0: 0                       ;CONTAINS NO. OF CHAR.'S ON SCREEN
5273  024232  000000          CHRCT1: 0
5274  024234  000000          CHRCT2: 0
5275  024236  000000          CHRCT3: 0
5276  024240  000000          HOLDS0: 0                       ;SET ON ERROR DETECTION, CLR'D VIA SW15
5277  024242  000000          HOLDS1: 0
5278  024244  000000          HOLDS2: 0
5279  024246  000000          HOLDS3: 0
5280  024250  000000          PATERN: 0
5281          024262                  .=.+10
5282  024262  000000          BLKERR: 0
5283          024274                  .=.+10
5284  024274  000000          AUTSW0: 0
5285  024276  000000          AUTSW1: 0
5286  024300  000000          AUTSW2: 0
5287  024302  000000          AUTSW3: 0
5288  024304  000000          TRANF0: 0
5289  024306  000000          TRANF1: 0
5290  024310  000000          TRANF2: 0
5291  024312  000000          TRANF3: 0
5292  024314  000000          RECTR0: 0
5293  024316  000000          RECTR1: 0
5294  024320  000000          RECTR2: 0
5295  024322  000000          RECTR3: 0
5296  024324  000000          STOP0:  0                       ;SET TO STOP CONTINOUS TRANSFER
5297  024326  000000          STOP1:  0
5298  024330  000000          STOP2:  0
5299  024332  000000          STOP3:  0
5300  024334  000000          ERFLG0: 0
5301  024336  000000          ERFLG1: 0
5302  024340  000000          ERFLG2: 0
5303  024342  000000          ERFLG3: 0
```

```
 5304  024344  000000  INTRN0: 0                       ;SET ON DL11 TRANS. INITIALIZATION
 5305  024346  000000  INTRN1: 0
 5306  024350  000000  INTRN2: 0
 5307  024352  000000  INTRN3: 0
 5308  024354  000000  RESTR0: 0                       ;CONTAINS 1ST FREE LOCATION AFTER
 5309  024356  000000  RESTR1: 0                       ;THE USER DATA
 5310  024360  000000  RESTR2: 0
 5311  024362  000000  RESTR3: 0
 5312  024364  000000  XFER0:  0
 5313  024366  000000  XFER1:  0
 5314  024370  000000  XFER2:  0
 5315  024372  000000  XFER3:  0
 5316  024374  000000  BUSY0:  0                       ;SET IF DOING CONTINOUS TRANSFERS
 5317  024376  000000  BUSY1:  0
 5318  024400  000000  BUSY2:  0
 5319  024402  000000  BUSY3:  0
 5320  024404  000000  BUF0R0: 0
 5321  024406  000000  BUF1R0: 0
 5322  024410  000000  BUF2R0: 0
 5323  024412  000000  BUF3R0: 0
 5324  024414  000000  BUF0R1: 0
 5325  024416  000000  BUF1R1: 0
 5326  024420  000000  BUF2R1: 0
 5327  024422  000000  BUF3R1: 0
 5328  024424  000000  START:  0
 5329  024426  000000  DISPSW: 0
 5330  024430  000000  SELECT: 0
 5331  024432  000000  SHFTSW: 0
 5332  024434  000000  SHFPRT: 0
 5333  024436  000000  SAVOLD: 0                       ;TEMPORARY STORAGE
 5334  024440  000000  TEMP1:  0                       ;TEMPORARY STORAGE
 5335  024442  000000  TEMP2:  0                       ;TEMPORARY STORAGE
 5336  024444  000000  MESPRT: 0                       ;SOFTWARE PRINT INHIBIT SW.
 5337  024446  000000  MESPT1: 0
 5338  024450  000000  SAVEPC: 0                       ;TEMPORARY STORAGE OF 'PC'
 5339  024452  000000  SAVPSW: 0                       ;TEMPORARY STORAGE OF 'PSW'
 5340  024454  000000  SAV2PC: 0
 5341  024456  000000  SAV2SW: 0
 5342  024460  000000  KSTOR1: 0                       ;PERMANENT STORAGE
 5343  024462  000000  KSTOR2: 0                       ;PERMANENT STORAGE
 5344  024464  000000  KSTOR3: 0                       ;PERMANENT STORAGE
 5345  024466  000000  KSTOR4: 0                       ;PERMANENT STORAGE
 5346  024470  000000  KSTOR5: 0
 5347  024472  000000  KSTOR6: 0
 5348
 5349                  ;HERE STARTS A '1200' WORD 'VT' BUFFER USED FOR ALL
 5350                  ;VT DISPLAY TESTS.  DATA TO BE DISPLAYED IS ASSEMBLED
 5351                  ;IN EACH INDIVIDUAL TEST AND STORED HERE.
 5352
 5353  024474     031  VTBUFF: .BYTE   EOS
 5354  024475     031          .BYTE   EOS
 5355
 5356
 5357          000001          .END
```

```
ASEMBL= 104015          1259#   1550    1555    1558    1570    1587    1604    1617    1660    2999    3713
AUDIO = 000200          1448#   3030    4425    4594
AUTOLL  012504          3429    3436#
AUTOST  013272          3482    3553#
AUTOUT  013304          3436    3559#
AUTSWC  024274          3428    3459*   3516*   3554*   3559*   3560*   3561    3563*   3564    5284#
AUTSW1  024276          5295#
AUTSW2  024300          5286#
AUTSW3  024302          5287#
AVECTR  024130          1646*   2955*   2963*   3010*   3075*   3787    4375*   4376*   5240#
BINDEC= 104024          1266#   3180    3183    3186    3189    3192    3195    3198    3201    3206    3209    3212    3215
                        3220    3223    3226    3229
BLANK = 000361          2446#   2427    2432    4304
BLINK = 000362          1447#   3417    4300
BLKERR  024262          3457*   3543*   3634    3637*   5282#
BOLD  = 000364          1445#   4302
BUFOR0  024404          3118*   3317    3371*   5320#
BUFOR1  024414          3119*   3318    3372*   5324#
BUF1R0  024406          3121*   3332    3382*   5321#
BUF1R1  024416          3122*   3333    3383*   5325#
BUF2R0  024410          3159*   3347    3393*   5322#
BUF2R1  024420          3160*   3348    3394*   5326#
BUF3R0  024412          3142*   3362    3404*   5323#
BUF3R1  024422          3143*   3363    3405*   5327#
BUSY0   024374          3446    3456*   3472    3487    3494*   3646*   5316#
BUSY1   024376          5317#
BUSY2   024400          5318#
BUSY3   024402          5319#
CAR   = 000010          1466#   1828    1841    1854    1867    1878
CHKFK0  007646          2921    2927#
CHKFK1  007704          2931    2933    2935    2936#
CHKFK2  007730          2940    2943#   2978    2981
CHKFK4  010004          2950    2952    2957#
CHRCNT  024146          3759*   3770*   3771    3790    3792*   4346*   4351    4358*   5247#
CHRCT0  024230          3179    3464*   3503    3505*   3508    3510*   5272#
CHRCT1  024232          3191    5273#
CHRCT2  024234          3205    5274#
CHRCT3  024236          3219    5275#
CHREND  020725          2969    4813#
CHRERR  010100          2938    2976#
CHRTAB  020464          2915    4619#
CKHALT  017520          4427    4435#   4450    4480    4527
CLRFKV  016710          1647    2660    2677    2693    2709    2723    2972    4268#
CLRFLD= 000360          1452#   4308
CLRVTV  016774          1648    2137    2159    2183    4289#
CNTRIT  020653          2965    4762#
CNTRLA  023001          3786    5101#
CNTRLC  022774          1659    5100#
CONTRL  020640          2959    4751#
CORSIZ  001702          1488    1492#
CRLF    023005          3801    4547    5103#
CSR   = 000000          1462#
DAR   = 000002          1463#   1962    1982    2001
DECOUT  016620          4237    4241#
DECPNT  016642          4231    4249#
DEC1    016632          4242    4245#
```

```
                        4270*
FKLDA   001472          1393*   2877    2889*
FKLDB   001470          1392*   2883    2890*   4189
FKLOGI  006236          1711    2529    2544*   2847
FKLVL   001502          1397*   2668*   2684*   2700*   4193    4261*   4270    4271*
FKREPT  010126          1717    2995*   3010
FKRPTA  010144          2996    3001*
FKRPTB  010160          3000    3005*   3007
FKT1    006246          2540    2552*   4598
FKT10   006504          2638*
FKT11   006532          2651*
FKT12   006566          2666*
FKT13   006640          2682*
FKT14   006714          2698*
FKT15   006770          2715*
FKT16   007040          2727    2735*
FKT17   007072          2749*
FKT2    006262          2556    2563*
FKT20   007120          2761*
FKT21   007166          2779*
FKT22   007254          2802*
FKT23   007320          2820*
FKT24   007372          2781    2841*
FKT3    006314          2576*
FKT4    006342          2587*
FKT5    006362          2598*
FKT6    006412          2611*
FKT7    006446          2625*
FKOCSR  001334          1324*   3174*   3426    3439    3513*
FKODAT  001336          1325*   3443
FKOINT  001340          1326*   3106*
FKOLDA  001332          1323*
FKOLDB  001330          1322*   1510    1563    4178
FKOLVL  001342          1327*   3107*
FK1CSR  001350          1333*   3175*
FK1DAT  001352          1334*
FK1INT  001354          1335*   3108*
FK1LDA  001346          1332*
FK1LDB  001344          1331*   4181
FK1LVL  001356          1336*   3109*
FK2CSR  001364          1342*   3165*
FK2DAT  001366          1343*
FK2INT  001370          1344*   3153*
FK2LDA  001362          1341*
FK2LDB  001360          1340*   4184
FK2LVL  001372          1345*   3154*
FK3CSR  001400          1351*   3148*
FK3DAT  001402          1352*
FK3INT  001404          1353*   3136*
FK3LDA  001376          1350*
FK3LDB  001374          1349*   4187
FK3LVL  001406          1354*   3137*
FLGERR  012412          3416*   3441    3527    3595    3603    3608    3611
FREPT0  010234          3015    3020*
FREPT1  010260          3024    3028*
FREPT2  010330          3032    3035    3037    3040*
```

```
FRJMPC  024160          4208*  4210*  4221   5252*
GETMAR  020042          4509   4516*
GETREG= 104003          1249*  3861   3898   4038   4055   4382   4466   4526
HOLDS0  024240          1651*  3432   3463*  3659   3662*  3664*  3837   4070*  4072   5276*
HOLDS1  024242          5277*
HOLDS2  024244          5278*
HOLDS3  024246          5279*
ICOUNT  020402          1740*  2072*  2128*  2538*  2774*  4583   4596*
INBUF   014600          3758   3760*  3811*  4327
INITAC  002232          1563*
INITB   001726          1496*  1646   5240
INITBA  002136          1241   1541*  1614
INITC   002150          1497   1544*
INITCB  002436          1507   1560   1562   1613*
INITD   002456          1543   1619*
INITSF  004752          2075   2087   2103   2116   2202*  2227
INITTY  001624          1239   1478*
INPUTA  014364          3761*  3762   3776   3791
INPUTB  014416          3769*  3797
INTCHR  015170          2270   2297   2337   3911*
INTRN0  024344          3521   3523*  5304*
INTRN1  024346          5305*
INTRN2  024350          5306*
INTRN3  024352          5307*
INTSWH  024152          2267*  2290*  2336*  3911   3918*  5249*
JMPWOR  012510          3431   3437*
KSTOR1  024460          2289*  2301*  2919*  2945   2967*  3311*  3322   3326*  3337   3341*  3352   3356*  3367
                        3714*  3717*  3719   3724*  3725   3726*  3729   4034   5342*
KSTOR2  024462          5343*
KSTOR3  024464          2291*  2302*  4478*  4504*  4510*  4514*  4523*  4539*  4558*  4560   5344*
KSTOR4  024466          3883*  3896*  4374*  4379   5345*
KSTOR5  024470          3681*  3688*  4497*  4514   5346*
KSTOR6  024472          4496*  4510   5347*
LDCHAR  015226          3912   3919*
LDCHEB  015304          3936*  3975
LDCHRO  015232          3921*  3939
LDCHR1  015260          3920   3929*
LDCHR2  015464          3967   3969   3973*
LDFKVT  016654          2653   2717   2862   2920   3014   4259*
LDLINE= 104010          1254*  4301   4303   4305   4307
LDUNIT  016432          4180   4183   4186   4189*
LDVTVT  016740          2131   2149   2170   4278*
LINKOD  012500          3433   3435*
LOOP12  003510          1877*  1885
LOOP16  003704          1932*  1939
LOOP24  004102          1999*  2007
MASK    015166          3682*  3691   3695*  3884*  3890   3895*  3900*
MAUFK   024037          1584   5225*
MBELL   023750          4610   5213*
MDLVA   024075          1601   5231*
MED1    024002          1554   5218*
MEMSIZ  024140          1493*  5244*
MESPRT  024444          1738*  2537*  4494*  4541   4545*  5336*
MESPT1  024446          1739*  4491   4493*  5337*
MES1    021105          2858   4903*
MESIA   021160          1557   4911*
```

```
MES10   022077          2523    5008#
MES11   022251          2525    5036#
MES13   022370          4475    5051#
MES14   022433          3009    5057#
MES15   022506          2998    5065#
MES16   022526          4501    5069#
MES17   022571          4495    5076#
MES18   022655          4339    5085#
MES19   022704          4336    5089#
MES2    021240          2545    4920#
MES20   023013          1753    5109#
MES21   023037          4108    5114#
MES22   023077          2954    5121#
MES23   023143          2962    5129#
MES24   023204          2265    5135#
MES25   023240          2287    5140#
MES26   023270          2314    5144#
MES27   023315          2352    5148#
MES3    021263          1629    4924#
MES30   023341          2397    5152#
MES31   023371          2423    5157#
MES33   023421          2447    5162#
MES34   023445          2463    5166#
MES35   023465          2478    5169#
MES36   023512          2496    5173#
MES37   023544          2073    2101    5179#
MES38   023555          2085    2114    5182#
MES39   023566          2317    5185#
MES4    021312          2914    4929#
MES40   023611          2321    5190#
MES41   023635          2325    5195#
MES42   023661          2329    5200#
MES43   023711          3836    5205#
MES44   023731          3829    5209#
MES5    021376          1616    4939#
MES5A   022126          4215    5014#
MES6    021506          2785    4953#
MES6A   021534          2804    4957#
MES6B   021562          2823    4961#
MES6C   022150          4219    5018#
MES7    021610          2191    2843    4966#
MES8    021634          3712    4972#
MES80   022157          3234    5020#
MES9    021651          3076    3077    3080    3082    4976#
MFCP    021440          1549    4946#
MFKP    024033          1567    5224#
MONITR  002526          1627    1637#   1668    3781    3807    4222    4340
MOVBLK  005716          2431#   2435
MOVEIT  015112          3886#   3897
MOVE1   014102          3685#   3697
MOVE2   014136          3689    3695#
MRJNN1  014750          1565*   1569    1574*   1579*   1586    1591*   1598*   1603    1609*   3830*   3831*   3833    3844#
MTRSWH  024166          1544    1649*   5255#
MUTP    024026          1582    5223#
MXORH   023752          4567    5214#
NOP   = 000240          1214#
```

```
NOTTY   001634          1240    1480#
NULCT0  024210          3498*   3534*   5264#
NULCT1  024212          5265#
NULCT2  024214          5266#
NULCT3  024216          5267#
NULL  = 104022          1264#   1638    1754    3862    4204    4525
OCTPRT  015056          1301    3879#
OUTPTA  014434          3773#   3774    3795    3804
PATERN  024250          3430    3458*   3515*   3570*   3571*   3572    5280#
PC    =%000007          1203#   1516*   1527*   1573*   1590*   1625*   1645*   1647*   1648*   1680*   1748*   1755*   2075*
                        2076*   2087*   2090*   2103*   2104*   2116*   2119*   2131*   2137*   2149*   2159*   2170*   2183*
                        2208*   2227*   2247*   2253*   2270*   2292*   2297*   2304*   2319*   2323*   2327*   2331*   2337*
                        2341*   2542*   2546*   2653*   2660*   2677*   2693*   2709*   2717*   2723*   2862*   2920*   2972*
                        3013*   3014*   3112*   3116*   3140*   3157*   3238*   3241*   3242*   3247*   3248*   3253*   3254*
                        3259*   3260*   3264*   3265*   3270*   3271*   3276*   3277*   3282*   3283*   3288*   3289*   3293*
                        3294*   3299*   3300*   3305*   3306*   3422*   3434*   3441*   3514*   3524*   3527*   3529*   3539*
                        3546*   3548*   3590*   3595*   3597*   3603*   3608*   3611*   3644*   3665*   3700*   3721*   3723*
                        3821*   3824*   3843*   3853*   3879*   3926*   3928*   3972*   3981*   4015*   4065*   4071*   4076*
                        4195*   4237*   4244*   4247*   4264*   4272*   4283*   4293*   4319*   4328*   4353*   4360*   4371*
                        4403*   4405*   4415*   4479*   4505*   4511*   4515*   4524*   4570*
PIMP    024126          1619*   1641    5239#
POP1SP= 005726          1210#   3827    3841
POP2SP= 022626          1213#   2659    2675    2691    2707    2722    2973    3784    4207
PRELOA= 104011          1255#   2398    2424    2448    2464    2479    2497
PRINT = 104012          1256#   1546    1548    1553    1557    1566    1568    1581    1583    1585    1600    1602    1615
                        1628    1658    1666    1752    2190    2997    3711    3785    3800    3808    3828    3832    3835
                        4214    4218    4335    4338    4512    4543    4546    4566    4609
PRTERR  020120          4479    4505    4534#
PRTOCT= 104014          1258#   4216    4220    4548    4553    4559
PSW     001312          1313#   1461*   1743*   2151*   2670*   2686*   2702*   3093*   3176*   4094*   4228*   4263*   4268*
                        4282*   4289*   4318*   4587*
PUSH1S= 005746          1211#
PUSH2S= 024646          1212#
PWRFAL  016014          1227    4084#
PWRUP   016050          4092    4094#
QMARK   023007          1667    3809    5106#
RBUF0   001530          1412#   3577
RBUF1   001532          1413#
RBUF2   001534          1414#
RBUF3   001536          1415#
RCSR0   001520          1408#   3125*
RCSR1   001522          1409#   3128*
RCSR2   001524          1410#   3162*
RCSR3   001526          1411#   3145*
RECSW0  024170          3094    3466*   3544*   3591    3647*   5256#
RECSW1  024172          5257#
RECSW2  024174          5258#
RECSW3  024176          5259#
RECTR0  024314          3182    3461*   3538*   3631*   3675*   5292#
RECTR1  024316          3194    5293#
RECTR2  024320          3208    5294#
RECTR3  024322          3222    5295#
RECVAB  014022          3622    3669#
RECVAC  014034          3616    3675#
RECVBB  013626          3620    3629#
RECVBC  013644          3634#   3671
```

```
RECVDD  014004          3656    3660    3663*
RECVEE  014010          3658    3664*
RECVIT  013362          3289    3294    3300    3306    3577*
RECVON  013444          3581    3587    3591*
RECVO   011766          3098    3288*
RECVOA  013472          3592    3594    3599*
RECVOB  013540          3600    3605    3610    3613*
RECVOC  013676          3633    3642*   3670
RECVOD  013706          3589    3630    3645*
RECVOE  014016          3618    3627    3650    3665*
RECV1   012000          3100    3293*
RECV2   012012          3149    3299*
RECV3   012024          3132    3305*
RELOAD  012556          3447    3450*   3484    3653
REPORT  005126          2235    2243*
REPTER  010264          3027    3030*
REPTOK  010312          3026    3029    3036*
RESCNT  005472          2355*   2362
RESTR0  024354          3453*   3474    3476    3493*   3654    5308*
RESTR1  024356          5309*
RESTR2  024360          5310*
RESTR3  024362          5311*
RETURN  020406          1742*   2263*   2285*   2312*   2350*   2395*   2421*   2445*   2461*   2476*   2494*   2521*   2540*
                        2782*   2856*   2912*   3007*   3073*   3728*   4109    4588    4593*   4598*   4611
RINT0   001540          1416*   1531    1596    3098*
RINT1   001544          1418*   3100*
RINT2   001550          1420*   3149*
RINT3   001554          1422*   3132*
RLVL0   001542          1417*   3099*
RLVL1   001546          1419*   3101*
RLVL2   001552          1421*   3150*
RLVL3   001556          1423*   3133*
RUNEM   016000          4067    4069    4075*
RUNEXT  016006          4077*
RUNSWH  024150          1650*   1730*   4066    5248*
R0      =%000000        1196*   1509*   1517*   1521*   1528*   1530*   1539*   1564*   1575*   1578*   1592*   1595*   1610*
                        2089*   2118*   2152*   2153*   2167*   2175    2176    2206*   2215*   2223    2228*   2230    2243
                        2249    2301    2355*   2357*   2372*   2373    2375*   2376    2377*   2402*   2403*   2408*   2409*
                        2427*   2431*   2432*   2501*   2502    2503*   2508*   2509    2510*   2565*   2566*   2567    2599*
                        2600*   2612*   2614*   2652*   2655*   2669*   2672*   2685*   2688*   2701*   2704*   2787*   2791*
                        2806*   2810*   2825*   2829*   2859*   2879    2880*   2882*   2889    2942*   2948*   2980*   3034*
                        3039*   3041*   3094*   3095*   3096    3110*   3111    3114    3117*   3118    3119    3178*   3181*
                        3184*   3187*   3190*   3193*   3196*   3199*   3204*   3207*   3210*   3213*   3218*   3221*   3224*
                        3227*   3312    3317*   3327    3332*   3342    3347*   3357    3362*   3371    3377*   3382    3389*
                        3393    3399*   3404    3410*   3417*   3418*   3419*   3420*   3450*   3451*   3476*   3478*   3492*
                        3493    3506*   3511*   3512*   3580    3586*   3623*   3624*   3625*   3638*   3639*   3640*   3642*
                        3643*   3654*   3693*   3694*   3698*   3699*   3913*   3922*   3931*   3934*   4035*   4046*   4048*
                        4051*   4053    4084    4102*   4156*   4157*   4160*   4161*   4162    4178*   4181*   4184*   4187*
                        4191    4243*   4246*   4300*   4302*   4304*   4306*   4308*   4389*   4391*   4428*   4452    4454*
                        4496
R1      =%000001        1197*   1489*   1490    1492*   1493    1669    1670*   1673*   1678*   1765*   1766*   1776*   1777*
                        1786*   1787*   1788    1816*   1817*   1818    2166*   2178*   2220*   2221*   2222*   2234*   2236
                        2238*   2240*   2241    2354*   2360*   2361    2401*   2406*   2428*   2434*   2500*   2506*   2716*
                        2719*   2724    2728    2788*   2789*   2807*   2808*   2826*   2827*   2860*   2885    2886*   2888*
                        2890    2915*   2936    2937    2951    2959    2965    2969    3022*   3025    3028    3041    3234*
                        3235    3236    3313    3318*   3328    3333*   3343    3348*   3358    3363*   3372    3376*   3383
```

```
SCOPEC  020272          1290    4576*
SCOPEF  020404          1747*   3727*   4429    4455    4551    4583    4585*   4589*   4597*
SCOPES  020350          4582    4584    4589*
SCOPI   020312          4531*   4604
SELECT  024430          2074*   2086*   2102*   2115*   2216    5330*
SERFKC  011626          3106    3241*
SERFK1  011642          3108    3247*
SERFK2  011656          3153    3253*
SERFK3  011672          3136    3259*
SERVAA  012664          3445    3468*
SERVAB  012706          3474*   3491
SERVBB  012770          3477    3479    3492*   3663
SERVBC  013030          3499*   3535
SERVCC  013064          3502    3508*   3555    3565    3573
SERVCE  011326          3178*   3203    3217    3230
SERVDX  013116          3509    3515*
SERVFK  012444          3242    3248    3254    3260    3426*
SERVOA  012530          3440    3443*
SERVOB  012730          3469    3481*
SERVOC  013036          3490    3501*
SERVOD  013106          3435    3449    3467    3471    3473    3488    3500    3504    3507    3513*   3517
SERVOO  012514          3427    3439*
SETBF0  012036          3241    3264    3288    3311*
SETBF1  012102          3247    3270    3293    3326*
SETBF2  012152          3253    3276    3299    3341*
SETBF3  012222          3259    3282    3305    3356*
SETED   002700          1516    1527    1573    1590    1669*
SETFKA  016442          4191*   4194
SETLNE= 104007          1253*   2300
SETUPF  006174          2536*   2546    3719    3723
SETUPV  003026          1736*   1755    3721
SETUT0  016360          4170    4177*
SETUT1  016376          4172    4181*
SETUT2  016410          4174    4184*
SETUT3  016422          4176    4187*
SHFBAD  005074          2226    2234*
SHFCUR  006106          2505*   2511
SHFEND  021024          2056    4887*
SHFEOL  005536          2373*   2379
SHFLST  005032          2223*
SHFMES  020026          2073*   2085*   2101*   2114*   4513*
SHFPRT  024434          1746*   5332*
SHFTBF  020726          2022    2024    2039    2041    2167    2168    2202    2203    2205    4825*
SHFTSW  024432          1745*   2129*   2207*   4459    4489    4508    5331*
SHF32   005046          2227*   2250
SHF64   005060          2224    2230*
SHIFT   005004          2076    2090    2104    2119    2215*   2229    2242    2251
SP    =%000006          1202*   1275*   1276*   1277*   1278    1281*   1282*   1283*   1284*   1285    1482*   1639*   2136*
                        2157*   2182*   2252*   3311    3312*   3313*   3314*   3315*   3316*   3326    3327*   3328*   3329*
                        3330*   3331*   3341    3342*   3343*   3344*   3345*   3346*   3356    3357*   3358*   3359*   3360*
                        3361*   3373    3374    3375    3376    3377    3384    3385    3386    3387    3388    3395    3396
                        3397    3398    3399    3406    3407    3408    3409    3410    3418    3421*   3855    3856*   3881
                        3882*   3923*   3927*   4077*   4084*   4085*   4086*   4087*   4088*   4089*   4090*   4091    4095*
                        4096    4097    4098    4099    4100    4101    4102    4115    4116    4117    4118    4119*   4120*
                        4121*   4122*   4123*   4124*   4125*   4126*   4127*   4133    4134    4135    4136    4137    4138
                        4139    4140    4141    4142*   4143*   4144*   4145*   4154    4155*   4206    4208    4259    4260*
```

```
                        4278    4279*   4349*   4374    4375    4377*   4431    4457    4478    4504    4586    4590    4592*
                        4593
SPACE = 104016          1260*   4550    4555    4561
SPACEX  014326          3740*   3742*   3744*   4399*   4400*   4402*   4406*   4408*   4410*
SPCHR1  014450          3766    3768    3779*
SPCHR3  014532          3789    3796*
SPCHR4  014540          3798*
SPCHR5  014554          3772    3799    3803*
SRHEOL  005526          2371*   2382
SRHEOS  005634          2405*   2410
SRIDE = 000004          1464*   2025    2042    2059    2074
SRIDO = 000014          1468*   2102
SRODE = 000006          1465*   2086
SRODO = 000016          1469*   2115
SRRVFK  007450          2863    2869*
SRVFOB  007522          2878    2883*
STACK   024132          1482    1639    2136    2157    2182    5241*
START   024424          3096    3321*   3336*   3351*   3366*   3450    3496    3545    3578    3586    5328*
STCOD0  024220          3455*   3619    3626*   3648*   5268*
STCOD1  024222          5269*
STCOD2  024224          5270*
STCOD3  024226          5271*
STERR   013462          3595*
STOP0   024324          3448*   3454*   3651    5296*
STOP1   024326          5297*
STOP2   024330          5298*
STOP3   024332          5299*
STRIPN  017200          4328    4350*   4364
STUNIT  016320          1625    1645    3824    4169*
SWAPEM  024164          1620*   1624*   1640*   1644*   3819    5254*
SWPEXT  014720          3826    3835*
SWR     001324          1318*   1500    1503    1505    1512    1515    1523    1526    1534    1542    1656    2245    2780
                        2930    2934    3002    3004    3530    3582    3593    3613    3657    3669    3717    3839    4019
                        4021    4068    4426    4435    4449    4506    4516    4519    4537    4577    4581    4607
SWR0    001326          1319*
SW00 = 000001           1192*   3532    3584    3593    3615    3669
SW01 = 000002           1191*   3532    3584    3615
SW02 = 000004           1190*
SW03 = 000010           1189*
SW04 = 000020           1188*
SW05 = 000040           1187*
SW06 = 000100           1186*
SW07 = 000200           1185*
SW08 = 000400           1184*
SW09 = 001000           1183*
SW10 = 002000           1182*
SW11 = 004000           1181*   4021
SW12 = 010000           1180*   2245    3839    4068
SW13 = 020000           1179*   4506
SW14 = 040000           1178*   2930    4019
SW15 = 100000           1177*
SYSIT1  01046'          3093*
SYSTST  010344          1719    3073*   3075
TAGA    006560          2654    2659*
TAGAB   006630          2667    2675*
TAGAH   007274          2807*   2811
```

```
TAGB    006704          2683    2691#
TAGC    006760          2699    2707#
TAGFA   007010          2718    2722#
TAGFH   007210          2782    2783#
TAGH    007230          2788#   2792
TAGI    007344          2826#   2830
TAGVA   004544          2132    2136#
TAGVB   004410          2078    2092#
TAGVC   004502          2106    2121#
TAGVD   004624          2150    2156#
TAGVE   004672          2172#   2179
TAGVF   004716          2171    2181#
TAG1X   011112          3149#   3169
TAG2X   011246          3131    3168#
TAG3X   011256          3166    3170#
TEMP1   024440          2917*   2949    2956*   4016*   4025*   4091*   4095    4476*   4498*   4502*   4562*   5334#
TEMP2   024442          2918*   2957    2964*   3982*   3983*   4012*   4014*   4023*   4551*   4552*   4554    5335#
TERMTB  015520          3917    3989#
TESTSW  013760          3434    3657#
TESTX   014162          1721    3709#
TEST00  002750          1665    1685#
TEST01  002752          1687#
TEST02  002754          1689#
TEST03  002756          1691#
TEST04  002760          1693#
TEST05  002762          1695#
TEST06  002764          1697#
TEST07  002766          1699#
TEST10  002770          1701#
TEST11  002772          1703#
TEST12  002774          1705#
TEST13  002776          1707#
TEST14  003000          1709#
TEST15  003002          1711#
TEST16  003004          1713#
TEST17  003006          1715#
TEST20  003010          1717#
TEST21  003012          1719#
TEST22  003014          1721#
THEND = 104023          1265#   2293    2452    2468    2483    2527
TITLE   021027          1547    4894#
TKB     001316          1315#   3763
TKS     001314          1314#   3751    3761    4317*
TKSFLG  014330          1304    3749#
TOPC    024156          4206*   4209*   4217    5251#
TPB     001322          1317#   3739*   3775*   3866*   3868*   3871*   3894*   4404*
TPS     001320          1316#   3737    3773    3857    3869    3892    4413
TRANF0  024304          3185    3460*   3645*   3676*   5288#
TRANF1  024306          3197    5289#
TRANF2  024310          3211    5290#
TRANF3  024312          3225    5291#
TRANO   011706          3102    3264#
TRANOA  013162          3526    3530#
TRANOB  013232          3536    3540#
TRANOC  013262          3533    3541    3547#
TRAN1   011722          3104    3270#
```

```
TRAN2   011736          3151    3276*
TRAN3   011752          3134    3282*
TRNSMT  013130          3265    3271    3277    3283    3521*
TRNSW0  024200          3465*   3495*   3525    3542*   3677*   5260*
TRNSW1  024202          5261*
TRNSW2  024204          5262*
TRNSW3  024206          5263*
TSTA    014204          3710    3716*
TSTB    014214          3715    3719*
TSTNUM  024142          1744*   2541*   3725*   4428    4454    4549    4591*   5245*
TSTSHF  005102          2231    2236*
TSTTKS= 104017          1261*   4576
TSTTPS  017436          4403    4405    4413*
TSWAP   014632          3818*   4071    4076
TTYENB  017062          3013    3853    3879    3981    4015    4065    4315*   4371
TTYIN = 104013          1257*   3753    4326
TTYMTR  002644          1654    1658*
TTYMT1  002652          1657    1661*
TTYSWH  024154          1478*   1480*   1496    1613    1626    1653    1750    2995    3709    3749    4211    4315    4423
                        4447    5250*
TYPECL  015030          3865    3868*
TYPERA  014772          3857*   3858    3867    3873
TYPER1  015014          3860    3864*
TYPER2  015022          3866*
TYPFKR  017626          4424    4475*
TYPMAR  020076          4517    4523*
TYPMES  014754          1299    3853*
TYPT1   016570          4232*   4239
TYPT2   016574          4233*   4235
TYPVTR  017662          4448    4488*
UNITFG  024134          1542*   1618*   1619    1621    1623*   1643*   3823*   3825    3831    4169    4171    4173    4175
                        4177*   5242*
UNITNO  024136          1503*   1504*   1509    1521    1530    1556*   1564    1578    1595    3078    3086    3130    3168
                        3202    3216    4169    5243*
UNLINE= 000370          1444*   4306
UPCASE  020552          2951    4685*
VISEOL= 000212          1453*   2355    2373    2377    3966    3989    3991    3993    3995    3997    3999    4001    4003
VISEOP= 000214          1454*   3968    3990    3992    3994    3996    3999    4000    4002
VISEOS= 000231          1455*   2403    2409    2503    2510    3970    4004
VTALIN  005740          1699    2445*   2454
VTBLAN  005660          1697    2421*   2437
VTBUFF  024474          1891    1904    1916    1930    1945    2289    2372    2402    2501    3110    3113    3117    3120
                        3138    3141    3155    3158    3178    3181    3184    3187    3190    3193    3196    3199    3204
                        3207    3210    3213    3218    3221    3224    3227    3321    3336    3351    3366    3913    4046
                        4051    4156    4161    5353*
VTCAR   001504          1399*   1827*   1833*   1840*   1846*   1853*   1855*   1865*   1877*   1893*   2024*   2041*   2502*
                        2509*   4052*   4190
VTCHAR  005166          1687    2194    2263*   2274
VTCRLF  017350          4388    4391*
VTCSR   001510          1401*   1736    1828*   1832*   1841*   1845*   2046    2318*   2322*   2326*   2330*   2404*   2429*
                        2451*   2467*   2482*   2504*   2524*   2526*   2786*   2805*   2824*   2976*   3030*   3822*   4054*
                        4152*   4281*   4290*   4372*   4425*   4461    4500*   4503    4518*   4522*   4594*
VTCSR4=%000004          1208*   1736*   1763*   1765    1785*   1786    1791*   1798*   1800    1803*   1810*   1815*   1854*
                        1867*   1878*   1882*   1892*   1905*   1917*   1932*   1946*   1950*   1962*   1966*   1982*   1986*
                        2001*   2005*   2025*   2026    2029    2032*   2042*   2050*   2058*   2059    2062*   2138    2216*
VTCURS  006042          1705    2494*   2513
```

```
VTDUMP  017764          4499    4504#
VTEOL   005454          1693    2334    2350#   2384
VTEOS   005574          1695    2395#   2412
VTERA   017312          4380#   4390    4393
VTERR   017532          1741    4447#
VTERR1  017614          4460    4464#
VTERR2  017620          4463    4465#
VTER1   017336          4381    4387#
VTER2   017344          4389#
VTFOCU  005766          1701    2461#   2469
VTINT   001514          1403#   4278#   4291#
VTLOGI  003122          1685    1731    1750#   2846
VTLVL   001516          1404#   4280#   4291    4292#
VTMAR   001512          1402#   1737    4462    4464    4523
VTMARS=%000005          1209#   1737#   1776    1812    1816    1829    1842    1856    1868    1879    1895    1909    1920
                        1935    1951    1963    1983    2002    2043    2225    2232    2241
VTMD1A  005274          2292    2297#   2303
VTMD2A  005342          2318#
VTMD2B  005434          2319    2323    2327    2331    2336#
VTMES   017246          1289    4371#
VTMOD1  005226          1689    2285#   2294
VTMOD2  005322          1691    2295    2312#   2333
VTNORM  017736          4490    4500#
VTODD   006136          1707    2521#   2528
VTREST  003136          1755#   2195
VTRUN   003016          1709    1730#
VTSAR   001506          1400#   1891#   1893#   1904#   1906#   1916#   1918#   1930#   1933#   1945#   1947#   1960#   1980#
                        1999#   2022#   2039#   2056#   2130#   2148#   2168#   2175#   2202#   2218#   2525#   4053#   4162#
VTWORS  006014          1703    2476#   2484
VTOCAR  001410          1358#   1520    1577    4179
VTOCSR  001414          1360#   3088#   3172#
VTOINT  001420          1362#
VTOLVL  001422          1363#
VTOMAR  001416          1361#
VTOSAR  001412          1359#   3076#   3114#
VT1     003142          1742    1761#
VT1CAR  001424          1367#   4182
VT1CSR  001430          1369#   3089#   3173#
VT1INT  001434          1371#
VT1LVL  001436          1372#
VT1MAR  001432          1370#
VT1SAR  001426          1368#   3077#   3115#
VT10    003422          1852#
VT11    003452          1864#
VT12    003500          1875#
VT13    003540          1890#
VT14    003576          1903#
VT15    003634          1915#
VT16    003666          1929#
VT17    003730          1944#
VT2     003166          1774#
VT2CAR  001440          1376#   4185
VT2CSR  001444          1378#   3083#   3164#
VT2INT  001450          1380#
VT2LVL  001452          1381#
VT2MAR  001446          1379#
```

```
VT2SAR  001442          1377*   3082*   3156*
VT20    003772          1959*
VT21    004024          1972*
VT22    004032          1979*
VT23    004064          1991*
VT24    004072          1997*
VT25    004130          2013*
VT26    004136          2021*
VT27    004206          2038*
VT3     003206          1784*
VT3CAR  001454          1385*   4188
VT3CSR  001460          1387*   3081*   3147*
VT3INT  001464          1389*
VT3LVL  001466          1390*
VT3MAR  001462          1388*
VT3SAR  001456          1386*   3080*   3139*
VT30    004260          2055*
VT31    004312          2070*
VT32    004412          2099*
VT33    004504          2127*
VT34    004564          2147*
VT35    004640          2165*
VT36    004730          2189*
VT4     003236          1797*
VT5     003262          1809*
VT6     003322          1826*
VT7     003362          1839*
WAITSY= 104021          1263*   1799    1811    1894    1907    1919    1934    1949    1961    1981    2000    2023    2040
                        2057    2133    2134    2169    2177    2204    2219
WHY     017144          4331    4338*
WCRD2   017154          4342*   4355
WORST   013342          3486    3570*
WORST1  013350          3437    3571*
X       = 000025        1215*   2563    2564*   2576    2577*   2587    2588*   2598    2599*   2611    2612*   2625    2626*
                        2638    2639*   2651    2652*   2666    2667*   2682    2683*   2698    2699*   2715    2716*   2735
                        2736*   2749    2750*   2761    2762*   2779    2780*   2802    2803*   2920    2821*   2841    2842*
XASEMB  017104          1302    4326*
XASEM1  017116          4329*   4347
XASEXT  017150          4337    4340*
XBINDE  016546          1309    4228*
XBUF0   001570          1429*   3499*   3532*   3547*   3678*
XBUF1   001572          1430*
XBUF2   001574          1431*
XBUF3   001576          1432*
XCSR0   001560          1425*   3126*
XCSR1   001562          1426*   3129*
XCSR2   001564          1427*   3163*
XCSR3   001566          1428*   3146*
XDELAY  015540          1293    4012*
XDLAYL  015550          1305    4014*
XDLAY1  015566          4017*   4024    4026
XDLAY2  015630          4022    4027*
XDLAY3  015614          4018    4023*
XENDT   015732          1295    4064*
XEOSBF  015656          1294    4045*
XFER0   024364          3452*   3470*   3649    5312*
```

```
XFER1   024366          5313#
XFER2   024370          5314#
XFER3   024372          5315#
XGETRG  016176          1292    4133#
XINTC   001600          1433#   1532    1597    3102*
XINT1   001604          1435#   3104*
XINT2   001610          1437#   3151*
XINT3   001614          1439#   3134*
XLDLNA  015644          4035#   4037
XLDLNE  015632          1297    4032#
XLOOP   014260          3727#   3728
XLVL0   001602          1434#   3103*
XLVL1   001606          1436#   3105*
XLVL2   001612          1438#   3152*
XLVL3   001616          1440#   3135*
XNULL   017360          1307    4399#
XNULL1  017404          4404#   4407
XORERR  020434          4601    4605#
XORFLG  020462          4534    4579    4612#
XORHLT  020270          4536*   4564    4571#
XORT    020410          4580    4600#
XPRELD  016252          1298    4152#
XPRT    020230          4557#   4563
XPRTA   020236          4511    4515    4524    4556    4559#
XSAVRG  016122          1291    4115#
XSPACE  014276          1303    3737#   3741
XSTLNE  017024          1296    4300#
XTHEND  015726          1308    4063#   4073
XTTYIN  014350          1237    1300    3758#   3810
XWAITS  015476          1306    3981#
Y       = 000037        1216#   1774    1775#   1784    1785#   1797    1798#   1809    1810#   1826    1827#   1839    1840#
                        1852    1853#   1864    1865#   1875    1876#   1890    1891#   1903    1904#   1915    1916#   1929
                        1930#   1944    1945#   1959    1960#   1972    1973#   1979    1980#   1991    1992#   1997    1998#
                        2013    2014#   2021    2022#   2038    2039#   2055    2056#   2070    2071#   2099    2100#   2127
                        2128#   2147    2148#   2165    2166#   2189    2190#
.       = 024476        1219#   1224#   1226#   1229#   1234#   1236#   1238#   1274#   1479    1491    1545    1622    1664
                        1751    1767    1778    1789    1801    1813    1819    1830    1843    1857    1869    1880    1896
                        1909    1921    1936    1952    1964    1984    2003    2027    2030    2044    2047    2060    2139
                        2154    2155    2174    2180    2193    2237    2239    2244    2246    2268    2272    2298    2339
                        2359    2374    2568    2580    2591    2601    2603    2615    2617    2628    2632    2641    2644
                        2656    2658    2673    2674    2689    2690    2705    2706    2720    2725    2729    2738    2742
                        2753    2764    2768    2790    2794    2809    2813    2828    2832    2845    2865    2923    2944
                        2946    2966    2970    3017    3097    3475    3522    3562    3602    3607    3641    3652    3684
                        3720    3722    3738    3750    3752    3780    3783    3806    3812#   3820    3838    3840    3870
                        3885    3893    3925    3930    3937    3971    3984    4013    4020    4050    4106    4159    4212
                        4316    4333    4352    4359    4414    4436    4492    4535    4542    5034#   5222#   5236#   5281*
                        5283#
```

F%*X	1243#	2563	2576	2587	2598	2611	2625	2638	2651	2666	2682	2698	2715	2735	2749
	2761	2779	2802	2820	2841										
*A	1243#	1774	1784	1797	1809	1826	1839	1852	1864	1875	1890	1903	1915	1929	1944
	1959	1972	1979	1991	1997	2013	2021	2038	2055	2070	2099	2127	2147	2165	2189
*S	1242#	2563	2576	2587	2598	2611	2625	2638	2651	2666	2682	2698	2715	2735	2749
	2761	2779	2802	2820	2841										
V*X	1243#	1774	1784	1797	1809	1826	1839	1852	1864	1875	1890	1903	1915	1929	1944
	1959	1972	1979	1991	1997	2013	2021	2038	2055	2070	2099	2127	2147	2165	2189

ADD	1283	1672	1676	1679	2252	3421	3579	3724	3726	3856	3882	3923	3927	4077	4155
	4236	4260	4279	4349	4357	4377	4592								
ASL	1281	1662	4361	4362	4363	4521									
ASR	4343	4344	4345												
BEQ	1560	1562	1572	1589	1614	1627	1751	1767	1778	1789	1801	1813	1819	1830	1843
	1857	1869	1880	1896	1909	1921	1936	1952	1964	1984	2003	2030	2044	2233	2242
	2244	2246	2250	2407	2507	2556	2568	2580	2591	2603	2628	2632	2641	2644	2742
	2873	2878	2884	2933	2940	2944	2946	2970	2996	3026	3029	3037	3087	3169	3433
	3447	3471	3475	3482	3484	3486	3504	3533	3589	3602	3607	3616	3618	3630	3633
	3650	3652	3656	3660	3710	3750	3791	3797	3804	3826	3838	3865	3925	3930	3946
	3948	3950	3952	3954	3956	3958	3960	3962	3964	3967	3969	3971	4170	4172	4174
	4176	4316	4331	4333	4347	4355	4388	4490	4509	4517	4535	4565	4608		
BGE	3689														
BGT	1664	3535	3581	3741	4359	4563									
BHI	3720														
BHIS	2729	3509													
BIC	1282	1504	1551	1661	1766	1777	1787	1817	2221	2240	2566	2629	2980	2881	2886
	2887	3005	3006	3531	3560	3583	3614	3679	3764	4229	4290	4356	4500	4518	4520
	4540														
BICB	3691	3890													
BIS	2234	2238	2564	2577	2588	2613	2626	2630	2639	2736	2740	2750	2762	2771	2783
	2835	2882	2888	2891	2893	2947	2968	2976	3030	3042	3513	3563	3891	4262	4281
	4425	4522	4594												
BISB	3692	3831	4245												
BIT	2026	2029	2043	2046	2059	2236	2245	2579	2602	2616	2627	2631	2640	2741	2780
	2930	3561	3593	3601	3606	3669	3839	4019	4021	4068	4506	4516	4537	4577	4581
	4607														
BITB	3892														
BLE	3944														
BLOS	2725														
BLT	3203	3217	3536												
BMI	1506	2139	2738	2764	2794	2832	3427	3658	3766	3768	3772				
BNE	1279	1497	1518	1529	1540	1545	1576	1593	1611	1622	1642	1654	1674	1895	1939
	2007	2027	2047	2060	2154	2174	2179	2224	2226	2231	2237	2303	2359	2362	2374
	2379	2382	2435	2601	2615	2617	2656	2673	2689	2705	2720	2781	2790	2792	2809
	2811	2828	2830	2875	2931	2938	2950	2952	2958	2960	2966	2978	3024	3032	3079
	3097	3131	3237	3429	3431	3445	3469	3473	3488	3490	3502	3522	3526	3541	3562
	3585	3592	3594	3620	3622	3635	3670	3697	3780	3783	3789	3799	3806	3820	3840
	3860	3897	3912	3920	3937	3942	3984	4018	4020	4022	4024	4026	4037	4050	4067
	4069	4073	4106	4159	4194	4212	4239	4242	4352	4381	4407	4424	4448	4460	4492
	4507	4538	4542	4578	4580	4582									
BPL	2753	2768	2813	2871	2928	2935	3021	3440	3600	3738	3752	3762	3774	3858	3870
	3893	4235	4401	4409	4414	4427	4436	4450	4584						
BR	1479	1491	1543	1657	1668	2078	2106	2155	2180	2193	2194	2229	2235	2239	2251
	2268	2272	2274	2294	2295	2298	2333	2334	2339	2384	2410	2412	2437	2454	2469
	2484	2511	2513	2528	2529	2658	2674	2690	2706	2727	2845	2865	2895	2923	2975
	2981	3000	3017	3027	3035	3084	3166	3230	3290	3295	3301	3307	3443	3467	3477
	3479	3491	3500	3507	3517	3555	3565	3573	3587	3605	3610	3627	3641	3671	3684
	3715	3722	3776	3795	3810	3867	3873	3885	3933	3939	4013	4180	4183	4186	4337
	4364	4390	4393	4463	4499	4556	4604								
CLR	1480	1620	1623	1640	1643	1650	1651	1652	1738	1739	1745	1746	1747	1791	1803
	1815	1832	1833	1845	1846	1855	1882	1883	1918	1966	1986	2005	2032	2050	2062
	2129	2172	2206	2267	2290	2336	2354	2524	2537	2578	2670	2716	2770	2774	2786
	2798	2805	2807	2824	2826	2834	2859	2860	2861	2876	2916	2917	2918	2919	2941
	3011	3012	3038	3095	3176	3319	3320	3452	3453	3454	3455	3456	3457	3459	3459
	3460	3461	3462	3463	3464	3465	3466	3515	3516	3538	3542	3543	3646	3647	3648

```
        3661  3662  3727  3742  3759  3760  3818  3822  3932  3940  3974  4014  4052  4104  4152
        4177  4228  4263  4271  4282  4292  4318  4350  4372  4385  4399  4404  4410  4494  4589
CLRB    2357  3499
CMP     1561  1621  1641  1663  1788  1800  1812  1829  1842  1879  1895  1908  1935  1963  1983
        2002  2223  2225  2230  2232  2241  2243  2249  2361  2567  2724  2728  2872  2874  2943
        2951  2959  2965  2969  3078  3086  3096  3130  3168  3202  3216  3508  3532  3580  3584
        3615  3719  3771  3936  3943  4171  4173  4175  4193  4330  4332  4354  4583  4586  4635
CMPB    2373  2937  3025  3028  3236  3444  3468  3481  3483  3485  3489  3501  3540  3588  3621
        3629  3632  3765  3767  3779  3782  3788  3796  3798  3803  3859  3864  3945  3947  3949
        3951  3953  3955  3957  3959  3961  3963  3966  3968  3970  4169  4380  4387
COM     3040  3470  3965
COMB    3571
DEC     1517  1528  1539  1575  1592  1610  1673  2178  2358  2378  2381  2406  2434  2506  3505
        3534  3688  3696  3740  3792  3896  4049  4158  4346  4358  4406  4562
EMT     1246  1247  1248  1249  1250  1251  1252  1253  1254  1255  1256  1257  1258  1259  1260
        1261  1262  1263  1264  1265  1266
HALT    1280  1498  1501  1511  1514  1522  1525  1533  1541  1655  3001  3003  3716  4093  4213
        4437  4465  4568
INC     1624  1644  1730  2089  2118  2153  2173  2215  2302  2360  2451  2467  2482  2526  2600
        2614  2655  2672  2688  2704  2719  2789  2791  2808  2810  2827  2829  2869  2956  2964
        2967  2979  3033  3081  3083  3088  3089  3416  3448  3510  3523  3544  3559  3626  3631
        3636  3637  3645  3675  3676  3677  3770  3823  3918  3919  3924  3935  3973  3983  4023
        4025  4036  4063  4105  4233  4238  4383  4430  4456  4493  4536  4545  4552  4585
INCB    1574  1591  1609  4400  4408
JMP     1239  1240  1241  1285  1507  1665  1731  2195  2846  2847  3243  3249  3255  3261  3266
        3272  3278  3284  3322  3337  3352  3367  3435  3436  3437  3653  3663  3729  3781  3787
        3807  3975  4109  4222  4340  4480  4527  4588  4611
JSR     1516  1527  1573  1590  1625  1645  1647  1648  1755  2075  2076  2087  2090  2103  2104
        2116  2119  2131  2137  2149  2159  2170  2183  2227  2270  2292  2297  2319  2323  2327
        2331  2337  2546  2653  2660  2677  2693  2709  2717  2723  2862  2920  2972  3013  3014
        3112  3116  3140  3157  3241  3242  3247  3248  3253  3254  3259  3260  3264  3265  3270
        3271  3276  3277  3282  3283  3288  3289  3293  3294  3299  3300  3305  3306  3434  3441
        3527  3595  3603  3608  3611  3721  3723  3824  3853  3879  3981  4015  4065  4071  4076
        4237  4328  4371  4403  4405  4479  4505  4511  4515  4524
MOV     1275  1277  1284  1478  1481  1482  1488  1489  1493  1494  1495  1500  1503  1509  1510
        1512  1515  1520  1521  1523  1526  1530  1531  1532  1534  1535  1536  1537  1538  1542
        1552  1556  1563  1564  1571  1577  1578  1588  1595  1596  1597  1605  1606  1607  1608
        1618  1619  1639  1646  1649  1656  1669  1670  1671  1675  1677  1678  1736  1737  1740
        1741  1742  1743  1744  1763  1765  1776  1785  1786  1798  1810  1816  1827  1828  1840
        1841  1853  1854  1865  1867  1876  1877  1878  1891  1892  1893  1904  1905  1906  1916
        1917  1930  1931  1932  1933  1945  1946  1947  1950  1960  1962  1980  1982  1998  1999
        2001  2022  2024  2025  2039  2041  2042  2056  2058  2072  2073  2074  2085  2086  2101
        2102  2114  2115  2128  2130  2136  2148  2151  2152  2157  2166  2167  2168  2175  2182
        2202  2203  2205  2207  2216  2218  2220  2228  2263  2285  2289  2291  2301  2312  2319
        2322  2326  2330  2350  2356  2370  2371  2372  2395  2401  2402  2404  2408  2421  2428
        2429  2445  2461  2476  2494  2500  2501  2502  2503  2504  2508  2509  2510  2521  2525
        2536  2538  2539  2540  2541  2565  2570  2582  2599  2605  2612  2619  2652  2667  2668
        2669  2671  2683  2684  2685  2686  2687  2699  2700  2701  2702  2703  2755  2782  2787
        2796  2806  2815  2821  2825  2856  2877  2879  2883  2885  2889  2890  2912  2915  2955
        2963  3007  3010  3073  3075  3076  3077  3080  3082  3093  3094  3098  3099  3100  3101
        3102  3103  3104  3105  3106  3107  3108  3109  3110  3111  3113  3114  3115  3117  3118
        3119  3120  3121  3122  3123  3124  3125  3126  3127  3128  3129  3132  3133  3134  3135
        3136  3137  3138  3139  3141  3142  3143  3144  3145  3146  3147  3148  3149  3150  3151
        3152  3153  3154  3155  3156  3158  3159  3160  3161  3162  3163  3164  3165  3170  3171
        3172  3173  3174  3175  3178  3179  3181  3182  3184  3185  3187  3188  3190  3191  3193
        3194  3196  3197  3199  3200  3204  3205  3207  3208  3210  3211  3213  3214  3219  3220
```

```
            3221  3222  3224  3225  3227  3228  3234  3311  3312  3313  3314  3315  3316  3317  3318
            3321  3326  3327  3328  3329  3330  3331  3332  3333  3334  3335  3336  3341  3342  3343
            3344  3345  3346  3347  3348  3349  3350  3351  3356  3357  3358  3359  3360  3361  3362
            3363  3364  3365  3366  3371  3372  3373  3374  3375  3376  3377  3382  3383  3384  3385
            3386  3387  3388  3393  3394  3395  3396  3397  3398  3399  3404  3405  3406  3407  3408
            3409  3410  3450  3476  3493  3494  3495  3496  3498  3530  3545  3553  3554  3564  3577
            3578  3582  3586  3613  3654  3664  3681  3682  3683  3695  3714  3717  3725  3728  3739
            3758  3763  3769  3793  3855  3868  3871  3881  3883  3884  3895  3913  3914  3915  3916
            3917  3921  3938  3982  4012  4016  4033  4034  4046  4047  4051  4053  4054  4070  4084
            4085  4086  4087  4088  4089  4090  4091  4092  4094  4095  4096  4097  4098  4099  4100
            4101  4102  4115  4116  4117  4118  4119  4120  4121  4122  4123  4124  4125  4126  4127
            4133  4134  4135  4136  4137  4138  4139  4140  4141  4142  4143  4144  4145  4153  4154
            4156  4157  4161  4162  4178  4179  4181  4182  4184  4185  4187  4188  4189  4190  4191
            4192  4206  4208  4230  4231  4232  4259  4261  4268  4269  4270  4278  4280  4289  4291
            4317  4327  4329  4348  4374  4375  4379  4402  4428  4429  4431  4433  4434  4452  4453
            4454  4455  4457  4461  4462  4464  4475  4476  4477  4479  4495  4496  4497  4498  4501
            4502  4503  4504  4510  4514  4519  4523  4551  4557  4558  4587  4590  4591  4593  4600
            4601  4603  4606
MOVB        1565  1579  1598  2355  2375  2377  2403  2409  2427  2431  2432  2929  2942  2948  2980
            3002  3004  3022  3034  3039  3041  3235  3417  3418  3419  3420  3443  3451  3478  3492
            3506  3511  3512  3537  3547  3570  3572  3623  3624  3625  3638  3639  3640  3642  3643
            3678  3690  3693  3694  3698  3699  3775  3830  3866  3889  3894  3922  3931  3934  4035
            4048  4160  4243  4246  4300  4302  4304  4306  4308  4389  4391
NOP         1483  1484  1485  1486  1487  1499  1502  1513  1524  1761  1762  2048  2092  2121  2271
            2299  2338  2552  2553  4205
RESET       1637  1764  1775  1866  1948  1973  1992  2014  2071  2100  2141  2158  2554  2589  2642
            3074  4103  4203
ROL         1884  1938  2006  3685  3686  3687  3886  3887  3888
RTI         2892  2971  3043  3378  3389  3400  3411  3743  3754  3802  3834  3842  3863  3899  3985
            4027  4039  4056  4074  4078  4128  4146  4163  4240  4309  4334  4386  4411  4438  4467
            4595
RTS         1680  1748  2208  2247  2253  2304  2341  2542  3238  3422  3514  3524  3529  3539  3546
            3548  3590  3597  3644  3665  3700  3821  3843  3926  3928  3972  4195  4244  4247  4264
            4272  4283  4293  4319  4353  4360  4415  4570
SUB         1276  1492  4209  4210  4234  4376  4432  4458  4539
SWAB        2222
*RAP        1245
TST         1278  1490  1496  1505  1544  1559  1613  1626  1653  1750  1818  1856  1868  1920  1951
            2088  2117  2138  2176  2217  2555  2590  2643  2751  2763  2766  2767  2831  2932  2934
            2939  2945  2949  2957  2977  2995  3023  3031  3036  3428  3430  3432  3439  3446  3472
            3474  3487  3503  3521  3525  3591  3599  3619  3634  3649  3651  3655  3657  3659  3709
            3749  3790  3794  3819  3825  3837  3911  3929  3941  4017  4066  4072  4211  4241  4315
            4342  4351  4423  4426  4435  4447  4449  4459  4489  4491  4508  4534  4541  4564  4579
            4602
TSTB        2376  2737  2752  2793  2812  2870  2927  2936  3020  3426  3497  3617  3737  3751  3761
            3773  3805  3857  3869  3872  4392  4413
WAIT        2864  2922  3016
.ABS        1169
.ASCII      4894  4903  4911  4920  4924  4929  4939  4946  4953  4957  4961  4966  4972  4976  4984
            4996  5008  5014  5018  5020  5028  5036  5045  5051  5057  5065  5069  5076  5085  5089
            5094  5100  5101  5103  5106  5109  5114  5121  5129  5135  5140  5144  5148  5152  5157
            5162  5166  5169  5173  5179  5182  5185  5190  5195  5200  5205  5209  5213  5214  5218
            5223  5224  5225  5231
.BYTE       2399  2400  2425  2426  2449  2450  2465  2466  2480  2481  2498  2499  3844  3989  3990
            3991  3992  3993  3994  3995  3996  3997  3998  3999  4000  4001  4002  4003  4004  4619
            4620  4621  4622  4623  4624  4625  4626  4627  4628  4629  4630  4631  4632  4636  4637
```

```
                4638    4639    4640    4641    4642    4643    4644    4645    4646    4647    4648    4649    4650    4654    4655
                4656    4657    4658    4659    4660    4661    4662    4663    4664    4665    4666    4670    4671    4672    4673
                4674    4675    4676    4677    4678    4679    4680    4681    4685    4686    4687    4688    4689    4690    4691
                4692    4693    4694    4695    4696    4697    4698    4702    4703    4704    4705    4706    4707    4708    4709
                4710    4711    4712    4713    4714    4715    4716    4720    4721    4722    4723    4724    4725    4726    4727
                4728    4729    4730    4731    4732    4736    4737    4738    4739    4740    4741    4742    4743    4744    4745
                4746    4747    4751    4752    4753    4754    4755    4756    4757    4758    4759    4760    4761    4762    4763
                4764    4768    4769    4770    4771    4772    4773    4774    4775    4776    4777    4778    4779    4780    4781
                4782    4786    4787    4788    4789    4790    4791    4792    4793    4794    4795    4796    4797    4798    4802
                4803    4804    4805    4806    4807    4808    4809    4810    4811    4812    4813    4825    4826    4827    4828
                4829    4830    4831    4832    4833    4834    4835    4836    4837    4838    4839    4840    4841    4842    4843
                4844    4845    4846    4847    4848    4849    4850    4851    4852    4853    4854    4855    4856    4857    4858
                4859    4860    4861    4862    4863    4864    4865    4866    4867    4868    4869    4870    4871    4872    4873
                4874    4875    4876    4877    4878    4879    4880    4881    4882    4883    4884    4885    4886    4887    4888
                4893    4982    4983    4995    5006    5030    5031    5032    5035    5050    5353    5354
.ENABL          1169
.END            5357
.EVEN           4005    4814    5034    5222    5236
.LIST           1169    1242    1775    1785    1798    1810    1827    1840    1853    1865    1876    1891    1904    1916    1930
                1945    1960    1973    1980    1992    1998    2014    2022    2039    2056    2071    2100    2128    2148    2166
                2190    2564    2577    2588    2599    2612    2626    2639    2652    2667    2683    2699    2716    2736    2750
                2762    2780    2803    2821    2842
.MACRO          1242    1243
.NLIST          1169    1242    1775    1785    1798    1810    1827    1840    1853    1865    1876    1891    1904    1916    1930
                1945    1960    1973    1980    1992    1998    2014    2022    2039    2056    2071    2100    2128    2148    2166
                2190    2564    2577    2588    2599    2612    2626    2639    2652    2667    2683    2699    2716    2736    2750
                2762    2780    2803    2821    2842
.REM               1
.REPT           1220
.TIT_E          1170
.WORD           3549    3666    4253


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 *,DBVTAD.SEQ/NL:TOC/SOL/CRF/PAGNUM=DSKZ:DBVTAD.P11
 RUN-TIME:  5 10 2 SECONDS
 RUN-TIME RATIO:  198/18=10.9
 CORE USED:  14K  (27 PAGES)
```