VS60

VISUAL DISPLAY TEST

MD-11-DZVSD-B

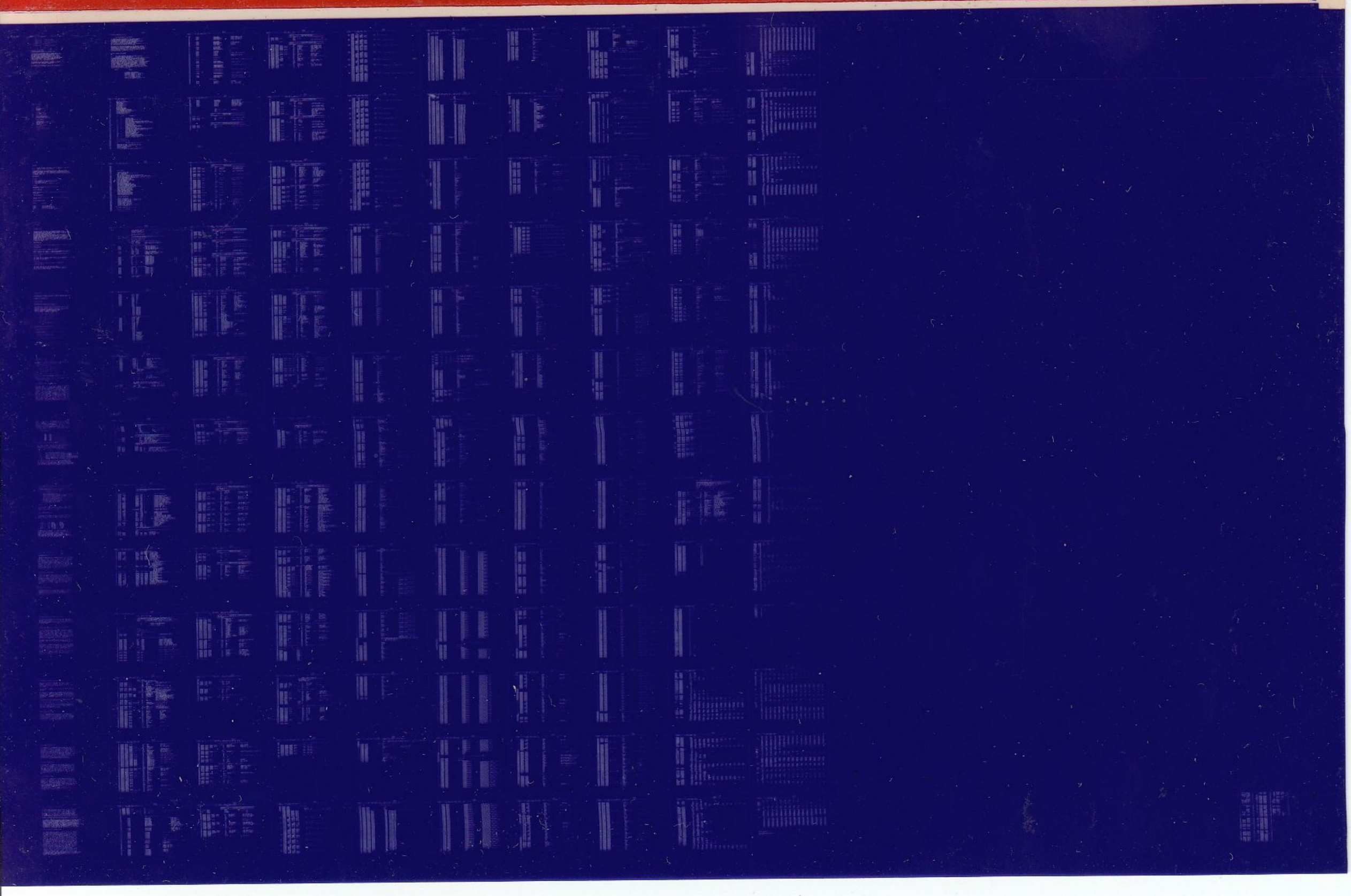
EP-DZVSD-B-DL-A

FICHE 1 OF 1

MAR 1977

digital

MADE IN USA



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

PRODUCT NAME: VS60 VISUAL DISPLAY TEST

DATE: JANUARY 1977

MAINTAINER: DIAGNOSTIC ENGINEERING

0.0 TABLE OF CONTENTS

1.0 ABSTRACT

2.0 REQUIREMENTS
2.1 EQUIPMENT
2.2 STORAGE

3.0 LOADING PROCEDURE

4.0 STARTING PROCEDURE

5.0 TEST PATTERN CONTROL
5.1 SWITCH REGISTER CONTROL
5.2 KEYBOARD CHARACTER CONTROL

6.0 ERROR REPORTING

7.0 MISCELLANEOUS
7.1 VS60 BUS/VECTOR ADDRESS MODIFICATION
7.2 XXDP/APT NOTES
7.3 POWER FAIL
7.4 SINGLE VS60 TESTING

8.0 EXECUTION TIME

9.0 PROGRAM TEST DESCRIPTIONS

10.0 LISTING

1.0 ABSTRACT

SEQ 0003

** DYNAMIC EXTURNAL STOP FRAME (W/27) IS DEPENDANT ON HARDWARE ECO'S (M7058 #5, M7054 #4 AND VT48 #7) **

THE PROGRAM PROVIDES THE OPERATOR WITH TWENTY TWO VISUAL FRAMES TO ADJUST OR VERIFY THE VISUAL OPERATION OF THE VS60 DISPLAY SYSTEM. NORMALLY EACH FRAME WILL CYCLE FOR ABOUT 5 SECONDS BEFORE ADVANCING TO THE NEXT FRAME. EACH VISUAL FRAME CAN BE SELECTED VIA SWITCH REGISTER OR KEYBOARD SELECTION.

2.0 REQUIREMENTS

2.1 EQUIPMENT

A. PDP-11 COMPUTER WITH AT LEAST 12K OF MEMORY
B. I/O TERMINAL (I.E. ASR33 TTY OR LK40)
C. VS-60 DISPLAY SYSTEM
D. ADDITIONAL VS-60 DISPLAY CONSOLE IF CONNECTED

2.2 STORAGE

THE PROGRAM OCCUPIES THE LOWER 8K OF MEMORY BUT REQUIRES 12K TO RUN.

3.0 LOADING PROCEDURE

NORMAL PROCEDURE FOR LOADING A BINARY PROGRAM INTO MEMORY SHOULD BE FOLLOWED.

4.0 STARTING PROCEDURE

LOADING ADDRESS 200 AND STARTING INITIALIZE THE SYSTEM, AND BEGIN TESTING.

5.1 SWITCH REGISTER CONTROL

SWITCH	FUNCTION
SW14=1	LOOP ON CURRENT TEST
SW09=1	STOP SUB-PICTURE MOTION
SW08=1	LOOP ON TEST IN SWR<4:0>
SW07=1	ENABLE KEYBOARD CONTROL (REF.5.2)

5.2 KEYBOARD CONTROL

STARTING THE TEST WITH SR7=1 WILL ENABLE KEYBOARD CONTROL.
KEYBOARD CONTROL IS AN AUXILIARY METHOD OF SELECTING THE TEST FRAME, LOOP ON A TEST FRAME, OR STOP-START FRAME MOTION. THE SWITCH REGISTER BITS OVERRIDE THE KEYBOARD CONTROL.
THE DIRECTORY FRAME PROVIDES THE OPERATOR WITH THE KEYBOARD LETTER AND SWITCH REGISTER VALUE FOR EACH TEST PATTERN. TO SELECT A TEST PATTERN, SIMPLY DEPRESS THE TEST LETTER ON THE CONSOLE KEYBOARD.
DEPRESS THE "RUB-OUT" KEY TO LOOP ON THE CURRENT TEST PATTERN.
DEPRESS THE "CR" KEY TO STOP MOTION. UNDEFINED TEST LETTERS WILL DISPLAY THE DIRECTORY FRAME. ALL OTHERS WILL HAVE NO EFFECT OTHER THAN TO RESUME PICTURE MOTION.

6.0 ERROR REPORTING

THE PROGRAM ONLY DISPLAYS VISUAL ERRORS AND DOES NOT REPORT ANY LOGIC ERRORS.

7.0 MISCELLANEOUS

7.1 VS60 BUS/VECTOR/PRIORITY ADDRESS MODIFICATION

MODIFY LOCATION 1242 ($VECT1) IF BASE VECTOR ADDRESS IS NOT 100320.
MODIFY LOCATION 1246 ($BASE) IF BASE BUS ADDRESS IS NOT 172000.

NOTE: A RESTART IS REQUIRED AFTER THE ABOVE ADDRESS MODIFICATION.

7.2 XXDP/APT NOTES

THE VISUAL TEST IS CHAINABLE UNDER XXDP IF 12K OR GREATER MEMORY IS AVAILABLE.
THE VISUAL TEST INCLUDES THE 'APT' SOFTWARE HOOKS, HOWEVER THEY HAVE NOT BEEN TESTED.

7.3 POWER FAIL

A POWER FAILURE WILL CAUSE THE PROGRAM TO BE RESTARTED.

7.4 SINGLE VS60 TESTING

THE VISUAL TEST DOES NOT TEST MULTIPLE VS60'S.
THE VISUAL TEST WILL UTILIZE THE SECOND CONSOLE IF CONNECTED.
THE "A" AND "U" FRAMES ARE USED TO VERIFY PROPER OPERATION BETWEEN THE TWO DISPLAY CONSOLES.

8.0 EXECUTION TIME

EXECUTION TIME IS APPROX. FOUR MIN. AN "END OF PASS" IS INDICATED BY A RETURN TO THE DIRECTORY FRAME. NO "END OF PASS" MESSAGE IS TYPED.

9.0 PROGRAM TEST DESCRIPTIONS

A = 01 Directory Frame

The sub-picture supplies the operator with a List of the Different Visual frames for his inspection.

This frame also includes a list of switch register values and keyboard control letters to select the visual frames. When a non-valid switch register value or keyboard key has been selected, the directory frame will be displayed. IF THE SECOND CONSOLE IS CONNECTED, THE OPERATOR SHOULD VERIFY THE "THIS IS CONSOLE 0" MESSAGE ON CONSOLE #0 AND THE "THIS IS CONSOLE 1" MESSAGE ON CONSOLE #1.

The frame is displayed by doing the following:

1. Point to x = 0 y = 1500
2. Enable console 1 intensity
3. Enter "character" mode and display inline text.
4. Display "STOP"
5. Display "JUMP ABSOLUTE" to the start of the frame.

B = 02 AstigMatism and Settling Time Frame

The frame will display points at individual bits at each x and y position register.

A floating one pattern used on each register followed by an accumulation pattern.

Bit 9 of x pos. Bit 9 of y pos.
Bit 8 of x pos. Bit 9 of y pos.
etc. etc.

Bits 9 and 8 of x pos. Bits 9 and 8 of y pos.
etc. etc.

C = 03 Short Term Drift Frame

The frame will display five points. The points will be displayed in each corner and the center of the screen. Each point actually consists of four "Display Point" instructions.

The point is generated by:

1. Positioning the x and y DAC at a cordinate.
2. IntensifYing the cordinate ONCE.
3. Do not intensify the point again for five (5) milliseconds.
4. Repeat 2 and 3 three more times.
5. If all the cordinates have not been displayed, update the cordinate and rePEAT 1 thru 4.

The C.P.U. cycle time is a factor in the 5 msec. delay routine. The current delay value (location "DELAY") is valid for a PDP-11/40 CPU type.

D = 04 Minor Axis Gain, Offset and Phase Frame

The frame consists of three square boxes with diagonal bisecting lines. The largest box encompASses the whole main screen viewing area. The second box, whose size is 100., is displayed in the right center area. The third box, whose size is 10., is below the second box. The boxes are drawn counter clockwise from the lower left corner. Upon completion the procedure is reversed and drawn clockwise from the lower left corner. When drawing the clockwise box the "Negative" polarity bit is set to enable adjustment of the "Offset" pot. Each box, upon completion, is segmented by a diagonal line from lower left to upper right and lower right to upper left ocorner. The frame also draws the same type box in the "menu" area. Because the "menu" is narrower than the high, the result is an rectangle in the menu area. In the lower center area, a series of four vectors 200 units long, are drawn from a common point. In the left center quadrant, ten vectors are drawn using "SHORT VECTOR" mode. Each of the vectors have a

length of eight units. After drawing the vector a 'RELATIVE POINT' is displayed two units away from the end of the vector. The "Y" coordinate is updated by two units and the "SHORT VECTOR" and "RELATIVE POINT" sequence is repeated. The visual result is a vertical "DOT-DASH" line. Included in the left quadrant is the Intensity Delay sub-picture. Eight vectors are drawn away from a "COMMON POINT" offset by ONE unit. The result will appear to be a square formed by the starting points of the vectors. Each vector has a length of 40 units.

VECTOR #	ORIGINATES AT X	Y
1	0354	1003
2	0354	1004
3	0353	1004
4	0352	1004
5	0352	1003
6	0352	1002
7	0353	1002
8	0354	1002

E = 05 Major Axis Offset and Vector Start Frame

The frame includes the minor axis gain frame plus two additional patterns. The first is used to adjust the vector starting point . The second pattern to adjust the major axis offset. The first pattern is drawn, in the upper quadrant, with the following manNEr:

Vector #	Direction
1	Positive Vertical Reference Vector.
2	Positive Horizontal Vector starting on VECTOR #1
3	Positive Horizontal Vector starting 1 unit RIGHT OF VECTOR #1
4	Negative Horizontal Vector starting on VECTOR #1
5	Negative Horizontal Vector starting 1 unit LEFT OF VECTOR #1
6	Positive Horizontal Vector start at the BOTTOM OF VECTOR #1
7	Negative Vertical Vector starting at the bottom OF VECTOR #1
8	Negative Horizontal Vector starting at the BOTTOM OF VECTOR #1

The second pattern draws, from a common point (x=1000, y=400), four pairs of vectors. The first of each paIR is drawn with the "y" axis being the major value. With the second using the "x" as the major. THE THIRD PATTERN CONSCISTS OF 10 PAIRS OF SHORT LENGTH VECTORS AND RELATIVE POINT'S DRAWN IN THE LEFT CENTER AREA.
AN 8 UNIT SHORT VERTICAL VECTOR IS DRAWN FOLLOWED BY A ONE UNIT RELATIVE POINT.

F = 06 Vector Length Gain, Convergence and Vector Linearity Frame

The pattern appears to be a series of horizontal lines being intersected by a diagonal line from upper left to lower right.

The picture is drawn by:

1. Draw an outer reference box
2. Starting from maximum, draw aN increasing negative length vector from an increasing "y" origin.
3. Starting from minimum x, draw a decreasing length vector from an increasing "y" origin.
4. Starting in the upper left edge, intensify a point at the intersection of #2 to #3 vector.
5. Starting in the upper left edge, intensify a decending vector that is over #4.
6. From center screen, using "BASIC" Vectors draw two intersecting "x" and "y" lines.

G = 07 Pincushion Frame

Using the "LONG" Vector instruction, display a "CROSS HATCH" visual pattern. The frame can be used to detect distortion in Vectors. From a distance of three feet, all vectors should appear straight with no vector curvature.

H = 10 OctagonS AND CIRCLES Frame

The purpose of the frame is to verify the endpoint matching of vectors. FIVE octagons are drawn from the center of the screen. The outer most octagon is drawn by using the "ABSOLUTE VECTOR" instruction from the point x = 530 y = 10.

Vector #	from	x-y	to	x-y
Vector 1	from	530-10	to	1250-10
Vector 2	from	1250-10	to	1770-530
Vector 3	from	1770-530	to	1770-1250
Vector 4	from	1770-1250	to	1250-1770
Vector 5	from	1250-1770	to	530-1770
Vector 6	from	530-1770	to	10-1250
Vector 7	from	10-1250	to	10-530
Vector 8	from	10-530	to	530-10

The FOUR concentric octagons are drawn by using the "LONG VECTOR" display instruction. The sizes are 377,177,77,7 respectively. Two more octagons with a size of 17 units are drawn at x = 300 y = 1000 an x = 1500 y = 1000. These two are drawn using the "BASIC SHORT" vector display instruction.
THREE CONCENTRIC CIRCLES ARE DRAWN USING ABSOLUTE VECTOR MODE.
EACH CIRCLE CONSCISTS OF 45. ABSOLUTE VECTORS. THE THREE CIRCLES HAVE A RADIUS OF 64., 128., AND 256. RESPECTIVELY.

I = 11 Scissoring and Vector Scaling Frame

The frame starts out by displaying a reference box around edge of the screen.

A VECTOR IS DRAWN FROM AN "ON-SCREEN" POSITION TO AN "OFF-SCREEN" POSITION. Another vector is drawn from the end of the previous vector back into the viewing area. This is repeated four times on each screen edge. The vectors should all terminate WITH NO bending or distortion. After all edges have been intersected, draw a large diamond that intersects each edge. The diamond and the vectors crossing the edges are the standard vecto length. To verify that vector scale operates proPerly, draw a square in the center of the screen. By changing the value of the "Vector Scale" register the box should increase in size. the vector scale is changed with the resulting picture being sixteen scaled boxes in the center of the screen.

J = 12 X and Y Dynamic Offset Frame

In this frame a square box is drawn in the center of the screen. The size is 1000 units and is drawn by the "Basic" vector instruction. The " offset" display instruction is used to modify the x and y offset registers. The box is moved from the center to the right side of the screen by changing the "offset instruction". The visual effect appears the viewing area is moving to the "left" side of the box. This pattern is repeated to the left, top and bottom edge respectIVELY. The number loaded into the offset registers range from 0 to 1400 with positive and negative polarity.

K = 13 Character Scale Frame

The frame function is to verify that character scale does change the size. To verify character scale, six characters, (the letters A, B, F, O, T and X) are displayed. each character startS with the largest to the smallest size on a common base line. A horizontal reference is drawn along the base of the characters.

L = 14 Character Quality and Character Rotate Frame

In this frame the message "The quick brown fox jumped over the lazy dogs" is displayed over the entire screen. By displaying the full screen of characters, the quality and distortion of the characters may be checked. Also included in the frame are rotated CHARACTERS. The rotated characters are displayed in the menu area.

M = 15 Character Set, Superscript, Subscript and Italic Frame

The frame displays all the displayable characters, special, italic, superscript and subscript. The first line consists of upper case letter (codes 100-137) and italic uppercase letters. The second line contains lower case letters (codes 140-177) and italic lower case letters. The third line contains numbers and punctuation (codes 40-77) and italic numbers and punctuation. the fourth line contains the special characters and italic special characters. These fouR lines are repeated in the lower half of the screen. Near the center of the screen a horizontal reference line is displayed.

The largest character scale is enabled and the letter "E" is displayed. This should appear on the base reference line. The code "super-script on" is enabled, followed by another "E".

The procedure is repeated three times with the result being four letter's of "E" with each having a reduced size and aN ascending y position. To verify the "superscript-off" function, the code "super-script off" followed by an ASCII "E" is sent. The procedure is repeated three times with the character increasing in size and decending in the y position.

The last "E" should be on the base reference line. The same procedure is repeated using the "subscript-on" and "subscript-off" codes except the characters should first decend with reducing in size followed by ascending and increasing in character size.

N = 16 Sync Speed and Character Terminate Frame

The patterns serves two FUNCTIONS. The first is to test character terminate. A diamond is displayed in the center of the screen with a message about the "SYNC" speed. The message is terminated by the value of '177' (a full dot matrix character).

The code #177 is loaded into the character terminate register and character terminate (character string escape) function is enabled.

The diamond is displayed using the "BASIC Vector" instruction. The message is displayed by entering "character" mode and doing a "display JSR" to the ASCII string. The text should be displayed and a "display POP and RESTORE" should occur after the code #177 is displayed. If "character terminate" fails to cause a "POP", a DIFFERENT message will be displayed reporting THE FACT.

The second purpose is to verify a visual change in the picture intensity when using NO SYNC, 40 cps sync and 30 cps sync.

The displayed message will indicate the different sync speeds.

When no sync is enabled the frame will appear bright and will have no flicker. When a sync speed of 40 is enabled, the frame will become dim. Upon selection of a sync speed of 30, the frame should appear to flicker. In each case, the frame appears different for each sync speed.

O = 17 Dash Lines and Blink Frame

This is a frame dedicated to the different line types and the ability to generate a blinking element. The type of line followed by two vectors of the same line type are displayed. The first is without blink enabled and the second is displayed with blink enabled. Visually the type of line is displayed followed by a non blinking line of the type followed by a blinking line of the type. This frame also used a "Display jump relative to loop" on the frame.

P = 20 Vector Length (Spray) Frame

The frame consists of "ABSOLUTE" vectors drawn from point 00 to another x,y point and a return vector to point 0,0. The first vector is drawn from point 0,0 to the maximum x and a y position of 1. Then a INVISIBLE vector to 0,0 is drawn. The third vector is drawn from point 0,0 to the maximum x and a y position of 3. This is repeated until the maximum y position has been displayed(45 DEG.). At that point the sequence is reversed IN that the x is the adjusted end point. The vector is drawn from point 0,0 to a value of x and the maximum value of y. A reference x and y vector is drawn at the right and top edge of the main screen. Each vector should terminate on the reference line. Even spacing should exist between the end of each vector. EVERY OTHER VECTOR WILL BE DISPLAYED.

Q = 21 Horizontal Phosphor Frame

In this frame, a reference box around the main screen perimeter is displayed. A band of intensified vectors are drawn to enable the operator to inspect phosphor surface. The band uses the "BASIC Vector" instruction by going the full value of y (path 2), delta x of 2 units (path0), negative full value of y (path 6) and a delta x of 2 units. This is repeated 50 times. The origin point of the band is updated via the "Point" instruction. The number of times the band is displayed before moving to the next position is controlled by the number loaded into the "TEMPA".

R = 22 Vertical Phosphor Frame

In this frame, a reference box around the main screen and menu perimeter is displayed. A band of intensified vectors are drawn thru the main screen and the menu screen to enable the operator to inspect the phosphor surface. The band uses "BASIC Vector" instruction by going the full value of x (path 0), delta y of 2 units (path 2), negative full value of x (path 4), and a delta y of 2 units. This is repeated 50 times. THE PROCESS IS THEN REPEATED AGAIN IN THE MENU AREA EXCEPT USING THE MAXIUM X MENU LENGTH (177).
The origin point of the band is updated via the "Point" instruction. The number of times the band is displayed before moving to the next position is controlled by the number loaded into the "TEMPA".

S = 23 Short Vector and Relative Point Frame

SEQ 0013

With this frame the operator can verify the correct selection of Relative point and short vectors. Four octagons are drawn in the four quadrants of the screen. Each octagon consists of an outer octagon drawn using the "short vector" instruction. Within each major octagon should be eight points at the intersecting vectors OF THE MAJOR OCTAGON. The "Relative point" instruction is used to display these points. A THIRD OCTAGON IS DISPLAYED USING THE "SHORT VECTOR" INSTRUCTION.

T = 24 GRAPHPLOT INCREMENT REGISTER TEST USING GRAPHPLOT X AND GRAPHPLOT Y

THE GRAPHPLOT INCREMENT REGISTER IS VERIFIED WITH A "SINE WAVE" PATTERN. TWO CYCLES OF A SINE WAVE ARE DISPLAYED IN GRAPHPLOT Y AND GRAPHPLOT Y MODES. THE AMOUNT OF INCREMENT BETWEEN POINTS IS A FUNCTION OF THE GRAPHPLOT INCREMENT REGISTER. AT THE END OF THE DISPLAY FILE IS A "DISPLAY STOP". UPON DETECTING THE DSTOP, A COUNTER IS DECREMENTED. UPON EXHAUSTION OF THE COUNTER, THE GRAPHPLOT INCREMENT REGISTER IS CHANGED. THE RESULT IS THE SINE WAVES WILL APPEAR TO EXPAND TO THE RIGHT, FOR GRAPHPLOT Y, AND TO THE TOP, FOR GRAPHPLOT X. ONLY THE LOWER THREE BITS OF THE INCREMENT REGISTER ARE VERIFIED WITH THIS PATTERN.

U = 25 Intensity Level and Lightpen Frame

The frame provides the operator with a method to visually check the eight different intensity levels. Points, Vectors and Characters are drawn using the different intensity levels. The frame also includes handling of "Light-pen" flags and "Light-Pen switches". An octagon is displayed in the upper riGht corner. Inside the octagon contain the X and Y axis values for the last "Light-Pen Hit". The state of the "Light-Pen switch" is also displayed within the octagon. In the lower right area a matrix of dots is used for a static test of the "Light-Pen field of View". The intensified dots are spaced four units apart. When the dots are detected by the "Light-Pen", the dot which a hit has occurred on will not be displayed. Below the dot matrix is aN octal readout reporting the hit count total. The center of the frame is bisected by a Horizontal Reference Line (Y=700). Nine vertical reference lines are drawn at 200 unit increments. The vertical lines are drawn below the Horizontal Reference Line are used to verify correct "X" pen hit position.
The lower left section contains vertical spacing test. Three parallel vectors are drawn with decreasing vertical spacing between the lines. The lower center area consists of a Variable Line Length Test. Twenty horizontal lines with increasing X length are drawn from a common X position. Both sections are used to test light pen selectivity. IF THE SECOND CONSOLE IS CONNECTED, VERIFY INDEPENDANT OPERATION OF THE X/Y AND PEN SWITCH READOUT FOR EACH CONSOLE. THE "FIELD OF VIEW" AND THE "HIT-COUNT" ARE THE ONLY DEPENDANT ELEMENTS.

V = 26 KEYBOARD CHARACTER ECHO LOOP

* MUST BE SELECTED BY THE OPERATOR *

THE FRAME PROVIDES A KEYBOARD TO VS60 SCREEN CHARACTER LOOP
TO VERIFY PROPER OPERATION OF THE CONSOLE KEYBOARD.
A MAXIMUM OF 1024. CHARACTERS CAN BE DISPLAYED BY THIS LOOP.
THE OPERATOR MAY ESCAPE THE LOOP, BY DEPRESSING THE "CTRL"
AND "C" KEYS, TO RETURN TO THE DIRECTORY FRAME.
UPON DECTECTION OF A KEYBOARD CHARACTER, THE CHARACTER'S OCTAL VALUE
AND THE CHARACTER ARE DISPLAYED ON THE SCREEN.

THE "SHIFT-OUT" CODE CAN BE ENTERED BY THE OPERATOR, HOWEVER THE
PROGRAM WILL NOT USE ANY KEYBOARD CODES GREATER THAN 37 OCTAL.
UPON ENTERING A "SHIFT-OUT" MODE, THE CHARACTER DISPLAYED FROM THE
CURRENT CHARACTER POSITION TO THE END OF THE LINE WILL APPEAR TO
BE AN UPSIDE DOWN "Y" CHARACTER. IN THE "SHIFT-OUT" MODE, THE
CHARACTER DISPLAYED HAS THE VALUE OF ZERO.

W = 27 DYNAMIC EXTERNAL STOP FRAME

** THIS FRAME VERIFIES VS60 ECO'S HAS BEEN INSTALLED **
(VT48 #7, M7054 #4 AND M7058 #5)
THIS FRAME VERIFIES PROPER OPERATION OF THE EXTERNAL DISPLAY STOP LOGIC.
A FRAME CONTAINING MOST OF THE VS60 INSTRUCTIONS IS DISPLAYED.
WHILE THE VS60 IS DISPLAYING THE FRAME, THE -11 CPU IS RANDOMLY
GENERATION A EXTERNAL DISPLAY STOP SIGNAL (EDSS) TO THE VS60.
AFTER AN "EDSS" HAS BEEN SENT, THE -11 WILL VERIFY THE DISPLAY PROGRAM
COUNTER REGISTER TO BE WITHIN AN EXPECTED RANGE. THE GENERATION OF
AN "EDSS" SHOULD CAUSE AN EXTURNAL STOP INTERRUPT. UPON DETECTING
AN "EDSS" INTERRUPT, A COUNTER IS DECREMENTED. IF THE COUNTER
DOES NOT GO TO 0, THE PROGRAM WILL ISSUE A "RESUME" TO THE VS60.
IF THE COUNTER BECOMES 0, THE PROGRAM WILL GO TO THE "END OF PASS"
AND RESTART THE PROGRAM. SEVEN DIFFERENT ERROR CONDITIONS WILL BE
VISUALY REPORTED WITH THIS SUB-TEST:

ERROR #	REASON
0	NO EXTERNAL STOP INTERRUPT
1	UNEXPECTED INTERRUPT TO VECTOR +4
2	UNEXPECTED INTERRUPT TO VECTOR +10
3	UNEXPECTED INTERRUPT TO VECTOR +14
4	D.P.C. OUT OF RANGE (TOO LOW)
5	D.P.C. OUT OF RANGE (TOO HIGH)
6	EXTERNAL STOP INTERRUPT BUT NO EXTERNAL STOP FLAG OR DISPLAY STOP FLAG.

10. LISTING

     13   BASIC DEFINITIONS
     18   OPERATIONAL SWITCH SETTINGS
     19   TRAP CATCHER
    (1)   STARTING ADDRESS(ES)
     21   ACT11 HOOKS
     23   APT PARAMETER BLOCK
     24   COMMON TAGS
    (2)   APT MAILBOX-ETABLE
    (1)   ERROR POINTER TABLE
     29   OPERATOR VARIABLE LOCATIONS
     38   VS-60 ADDRESSES AND INTERRUPT VECTORS
     66   INITIAL PROGRAM STARTUP ROUTINE
     69   INITIALIZE THE COMMON TAGS
    116   KEYBOARD SERVICE ROUTINE
    153   VS-60 INSTRUCTION SET
    265
    266   TEST   LETTER   DESCRIPTION
    267   ----   ------   -----------
    268
    270   T1   A   DIRECTORY FRAME
    277   T2   B   ASTIGMATISM AND SETTLING TIME
    284   T3   C   SHORT TERM DRIFT
    335   T4   D   MINOR AXIS GAIN, OFFSET AND PHASE ADJUSTMENT
    344   T5   E   MAJOR AXIS OFFSET AND VECTOR START POINT ADJUSTMENT
    356   T6   F   VECTOR LENGTH GAIN, CONVERGENCE AND VECTOR LINEARITY
    432   T7   G   PINCUSHION FRAME
    475   T10   H   OCTAGONS AND CIRCLES
    480   T11   I   SCISSORING AND VECTOR SCALING
    497   T12   J   OFFSET X AND OFFSET Y POSITION
    547   T13   K   CHARACTER SCALE FRAME
    557   T14   L   CHARACTER QUALITY AND CHARACTER ROTATE IN THE MENU
    576   T15   M   CHARACTER SET, SUPERSCRIPT, SUBSCRIPT AND ITALICS
    699   T16   N   SYNC SPEED AND CHARACTER TERMINATE TEST
    732   T17   O   DASH LINES AND BLINK
    737   T20   P   VECTOR SPRAY (LENGTH) TEST
    765   T21   Q   HORIZONTAL PHOSPHOR TEST
    781   T22   R   VERTICAL PHOSPHOR TEST
    808   T23   S   SHORT VECTOR AND RELATIVE POINT
    859   T24   T   GRAPHPLOT INCREMENT REGISTER TEST USING GRAPHPLOT X AND Y
    868   T25   U   INTENSITY LEVEL AND LIGHT PEN TEST
    906   T26   W   DYNAMIC EXT. DISPLAY STOP
    995
    997   END OF PASS ROUTINE
   1000
   1001   MANUAL INTERVENTION TEST
   1002
   1005   T27   V   KEYBOARD CHARACTER ECHO LOOP
   1056
   1059   SUBROUTINE FOR VERT. LIGHT PEN FIELD OF VIEW
   1067   SUBROUTINE FOR HORIZ. LIGHT PEN FIELD OF VIEW
   1077   LIGHT-PEN INTERRUPT SERVICE
   1167   DISPLAY SUB-ROUTINE
   1198   UPDATE OCTAL READOUT OF THE X-Y FOR LIGHT PEN HIT
   1221   X - Y POSITIONS FOR THE SHORT TERM DRIFT TEST

1253
1254
1255
1256
1257
1258 SUB-PICTURES
1259
1260 DIRECTORY SUB-PICTURE
1339 X AND Y POSITIONS FOR THE SETTLING TEST
1491 MENU 1 SUB-PICTURE
1616 DRAW 10 VERTICAL VECTORS IN THE LEFT CENTER AERA
1632 DRAW THE DELAY INTENSITY SUB-PICTURE IN THE LEFT CENTER AERA
1685 OCTAGONS USING LONG AND ABSOLUTE VECTORS (WIDTHS OF 7,77,177,377 AND 520)
1686 CIRCLES USING ABSOLUTE VECTORS (WIDTHS OF 64., 128., AND 256.)
1933 X AND Y OFFSET SUB-PICTURE
1951 SUPER AND SUBSCRIPT SUB-PICTURE
1986 SUPER AND SUBSCRIPT ASCII STRING
2000 SYNC SPEED SUBPICTURE
2038 DASH LINE SUB-PICTURE
2102 VECTOR LENGTH SUB-PICTURE
2127 HORIZONTAL PHOSPHOR SUB-PICTURE
2145 MAIN VERTICAL PHOSPHOR SUB-PICTURE
2162 MENU VERTICAL PHOSPHOR SUB-PICTURE
2225 SHORT VECTOR AND RELATIVE POINT SUB-PICTURE
2242 GRAPHPLOT INCREMENT SUB-PICTURE
2283 DATA STRING FOR A SINE WAVE
2296 SHORT TERM DRIFT SUB-PICTURE
2316 SCREEN SCISSORING SUB-PICTURE
2391 VECTOR SCALE SUB-PICTURE
2413 VECTOR STARTING SUB-PICTURE
2465 MAJOR AXIS OFFSET SUB-PICTURE
2527 CHARACTER SCALE SUB-PICTURE
2639 ROTATE CHARACTERS SUBPICTURE
2703
2704 LIGHT-PEN SUBPICTURE
2705
2708 POSITION THE OCTAGON
2715 DISPLAY ON CONSOLE #0 THE X-Y READOUT VALUE
2733 DISPLAY ON CONSOLE #1 THE X-Y READOUT VALUE
2752 DISPLAY HIT COUNT MESSAGE
2776 HORIZONTAL REF. LINE SECTION
2822 VERTICAL SPACEING SECTION
2841 VARIABLE HORIZ. LINE LENGTH
2862 INTENSITY LEVEL SECTION OF LIGHT PEN TEST
2986 DRAW OUTER REFERENCE BOX
3004
3005 KEYBOARD CHARACTER ECHO SUB-PICTURE
3025
3026 DYNAMIC EXT. STOP FRAME
3027
3261 SCOPE HANDLER ROUTINE

```
                        .TITLE  MAINDEC-11-DZVSDB  VS60 VISUAL DISPLAY TEST
                        ;*COPYRIGHT (C) 1976
                        ;*DIGITAL EQUIPMENT CORP.
                        ;*MAYNARD, MASS. 01754
                        ;*
                        ;*PROGRAM BY RAYMOND SHOOP
                        ;*
                        ;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
                        ;*PACKAGE (MAINDEC-11-DZQAC-C2), SEPT 14, 1976.
                        ;*
                        .SBTTL  BASIC DEFINITIONS

                        ;*INITIAL ADDRESS OF THE STACK POINTER *** 1100 ***
001100                  STACK=  1100
                        .EQUIV  EMT,ERROR       ;;BASIC DEFINITION OF ERROR CALL
                        .EQUIV  IOT,SCOPE       ;;BASIC DEFINITION OF SCOPE CALL

                        ;*MISCELLANEOUS DEFINITIONS
000011                  HT=     11              ;;CODE FOR HORIZONTAL TAB
000012                  LF=     12              ;;CODE FOR LINE FEED
000015                  CR=     15              ;;CODE FOR CARRIAGE RETURN
000200                  CRLF=   200             ;;CODE FOR CARRIAGE RETURN-LINE FEED
177776                  PS=     177776          ;;PROCESSOR STATUS WORD
                        .EQUIV  PS,PSW
177774                  STKLMT= 177774          ;;STACK LIMIT REGISTER
177772                  PIRQ=   177772          ;;PROGRAM INTERRUPT REQUEST REGISTER
177570                  DSWR=   177570          ;;HARDWARE SWITCH REGISTER
177570                  DDISP=  177570          ;;HARDWARE DISPLAY REGISTER

                        ;*GENERAL PURPOSE REGISTER DEFINITIONS
000000                  R0=     %0              ;;GENERAL REGISTER
000001                  R1=     %1              ;;GENERAL REGISTER
000002                  R2=     %2              ;;GENERAL REGISTER
000003                  R3=     %3              ;;GENERAL REGISTER
000004                  R4=     %4              ;;GENERAL REGISTER
000005                  R5=     %5              ;;GENERAL REGISTER
000006                  R6=     %6              ;;GENERAL REGISTER
000007                  R7=     %7              ;;GENERAL REGISTER
000006                  SP=     %6              ;;STACK POINTER
000007                  PC=     %7              ;;PROGRAM COUNTER

                        ;*PRIORITY LEVEL DEFINITIONS
000000                  PR0=    0               ;;PRIORITY LEVEL 0
000040                  PR1=    40              ;;PRIORITY LEVEL 1
000100                  PR2=    100             ;;PRIORITY LEVEL 2
000140                  PR3=    140             ;;PRIORITY LEVEL 3
000200                  PR4=    200             ;;PRIORITY LEVEL 4
000240                  PR5=    240             ;;PRIORITY LEVEL 5
000300                  PR6=    300             ;;PRIORITY LEVEL 6
000340                  PR7=    340             ;;PRIORITY LEVEL 7

                        ;*"SWITCH REGISTER" SWITCH DEFINITIONS
100000                  SW15=   100000
040000                  SW14=   40000
```

```
(1)             020000                  SW13=   20000
(1)             010000                  SW12=   10000
(1)             004000                  SW11=   4000
(1)             002000                  SW10=   2000
(1)             001000                  SW09=   1000
(1)             000400                  SW08=   400
(1)             000200                  SW07=   200
(1)             000100                  SW06=   100
(1)             000040                  SW05=   40
(1)             000020                  SW04=   20
(1)             000010                  SW03=   10
(1)             000004                  SW02=   4
(1)             000002                  SW01=   2
(1)             000001                  SW00=   1
(1)                                     .EQUIV  SW09,SW9
(1)                                     .EQUIV  SW08,SW8
(1)                                     .EQUIV  SW07,SW7
(1)                                     .EQUIV  SW06,SW6
(1)                                     .EQUIV  SW05,SW5
(1)                                     .EQUIV  SW04,SW4
(1)                                     .EQUIV  SW03,SW3
(1)                                     .EQUIV  SW02,SW2
(1)                                     .EQUIV  SW01,SW1
(1)                                     .EQUIV  SW00,SW0
(1)
(1)                                     ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)
(1)             100000                  BIT15=  100000
(1)             040000                  BIT14=  40000
(1)             020000                  BIT13=  20000
(1)             010000                  BIT12=  10000
(1)             004000                  BIT11=  4000
(1)             002000                  BIT10=  2000
(1)             001000                  BIT09=  1000
(1)             000400                  BIT08=  400
(1)             000200                  BIT07=  200
(1)             000100                  BIT06=  100
(1)             000040                  BIT05=  40
(1)             000020                  BIT04=  20
(1)             000010                  BIT03=  10
(1)             000004                  BIT02=  4
(1)             000002                  BIT01=  2
(1)             000001                  BIT00=  1
(1)                                     .EQUIV  BIT09,BIT9
(1)                                     .EQUIV  BIT08,BIT8
(1)                                     .EQUIV  BIT07,BIT7
(1)                                     .EQUIV  BIT06,BIT6
(1)                                     .EQUIV  BIT05,BIT5
(1)                                     .EQUIV  BIT04,BIT4
(1)                                     .EQUIV  BIT03,BIT3
(1)                                     .EQUIV  BIT02,BIT2
(1)                                     .EQUIV  BIT01,BIT1
(1)                                     .EQUIV  BIT00,BIT0
(1)
(1)                                     ;*BASIC "CPU" TRAP VECTOR ADDRESSES
```

```
(1)             000004                  ERRVEC= 4               ;;TIME OUT AND OTHER ERRORS
(1)             000010                  RESVEC= 10              ;;RESERVED AND ILLEGAL INSTRUCTIONS
(1)             000014                  TBITVEC=14              ;;"T" BIT
(1)             000014                  TRTVEC= 14              ;;TRACE TRAP
(1)             000014                  BPTVEC= 14              ;;BREAKPOINT TRAP (BPT)
(1)             000020                  IOTVEC= 20              ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
(1)             000024                  PWRVEC= 24              ;;POWER FAIL
(1)             000030                  EMTVEC= 30              ;;EMULATOR TRAP (EMT) **ERROR**
(1)             000034                  TRAPVEC=34              ;;"TRAP" TRAP
(1)             000060                  TKVEC=  60              ;;TTY KEYBOARD VECTOR
(1)             000064                  TPVEC=  64              ;;TTY PRINTER VECTOR
(1)             000240                  PIRQVEC=240             ;;PROGRAM INTERRUPT REQUEST VECTOR
  14            172000                          ABASE=172000
  15            100320                          AVECT1=100320
  16            000200                          APRIOR=200
  17
  18                                    .SBTTL  OPERATIONAL SWITCH SETTINGS
(1)                                     ;*
(1)                                     ;*      SWITCH                  USE
(1)                                     ;*      ------          --------------------
(1)                                     ;*        14            LOOP ON TEST
(1)                                     ;*         9            STOP SUB-PICTURE MOTION
(1)                                     ;*         8            LOOP ON TEST IN SWR<7:0>
  19                                    .SBTTL  TRAP CATCHER
(1)
(1)             000000                          .=0
(1)                                     ;*ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"
(1)                                     ;*SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS
(1)                                     ;*LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS
(1)             000174                          .=174
(1)  000174  000000                     DISPREG: .WORD  0               ;;SOFTWARE DISPLAY REGISTER
(1)  000176  000000                     SWREG:   .WORD  0               ;;SOFTWARE SWITCH REGISTER
(1)                                     .SBTTL  STARTING ADDRESS(ES)
(1)  000200  000137  001334                     JMP     @#BEGIN ;;JUMP TO STARTING ADDRESS OF PROGRAM
```

```
                                .SBTTL  ACT11 HOOKS

                                ;;******************************************************************
                                ;HOOKS REQUIRED BY ACT11
                                        $SVPC=.                 ;SAVE PC
                                        .=46
                                        $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
                                        .=52
                                        .WORD   0               ;;2)SET LOC.52 TO ZERO
                                        .=$SVPC                 ;; RESTORE PC
                                        .=1000
                                .SBTTL  APT PARAMETER BLOCK

                                ;;******************************************************************
                                ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
                                ;;******************************************************************
                                        .$X=.   ;;SAVE CURRENT LOCATION
                                        .=24    ;;SET POWER FAIL TO POINT TO START OF PROGRAM
                                        200     ;;FOR APT START UP
                                        .=44    ;;POINT TO APT INDIRECT ADDRESS PNTR.
                                        $APTHDR ;;POINT TO APT HEADER BLOCK
                                        .=.$X   ;;RESET LOCATION COUNTER
                                ;;******************************************************************
                                ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
                                ;INTERFACE SPEC.

                                $APTHD:
                                $HIBTS: .WORD   0       ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
                                $MBADR: .WORD   $MAIL   ;;ADDRESS OF APT MAILBOX (BITS 0-15)
                                $TSTM:  .WORD   20      ;;RUN TIM OF LONGEST TEST
                                $PASTM: .WORD   300     ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
                                $UNITM: .WORD   0       ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
                                        .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
```

```
                                .SBTTL  COMMON TAGS

                                ;;**********************************************************************
                                ;*THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS
                                ;*USED IN THE PROGRAM.

                        001100          .=1100
001100                  001100  $CMTAG:                 ;;START OF COMMON TAGS
001100  000000                          .WORD   0
001102  000                     $TSTNM: .BYTE   0       ;;CONTAINS THE TEST NUMBER
001103  000                     $ERFLG: .BYTE   0       ;;CONTAINS ERROR FLAG
001104  000000                  $ICNT:  .WORD   0       ;;CONTAINS SUBTEST ITERATION COUNT
001106  000000                  $LPADR: .WORD   0       ;;CONTAINS SCOPE LOOP ADDRESS
001110  000000                  $LPERR: .WORD   0       ;;CONTAINS SCOPE RETURN FOR ERRORS
001112  000000                  $ERTTL: .WORD   0       ;;CONTAINS TOTAL ERRORS DETECTED
001114  000                     $ITEMB: .BYTE   0       ;;CONTAINS ITEM CONTROL BYTE
001115  001                     $ERMAX: .BYTE   1       ;;CONTAINS MAX. ERRORS PER TEST
001116  000000                  $ERRPC: .WORD   0       ;;CONTAINS PC OF LAST ERROR INSTRUCTION
001120  000000                  $GDADR: .WORD   0       ;;CONTAINS ADDRESS OF 'GOOD' DATA
001122  000000                  $BDADR: .WORD   0       ;;CONTAINS ADDRESS OF 'BAD' DATA
001124  000000                  $GDDAT: .WORD   0       ;;CONTAINS 'GOOD' DATA
001126  000000                  $BDDAT: .WORD   0       ;;CONTAINS 'BAD' DATA
001130  000000                          .WORD   0       ;;RESERVED--NOT TO BE USED
001132  000000                          .WORD   0
001134  000                     $AUTOB: .BYTE   0       ;;AUTOMATIC MODE INDICATOR
001135  000                     $INTAG: .BYTE   0       ;;INTERRUPT MODE INDICATOR
001136  000000                          .WORD   0
001140  177570                  SWR:    .WORD   DSWR    ;;ADDRESS OF SWITCH REGISTER
001142  177570                  DISPLAY: .WORD  DDISP   ;;ADDRESS OF DISPLAY REGISTER
001144  177560                  $TKS:   177560          ;;TTY KBD STATUS
001146  177562                  $TKB:   177562          ;;TTY KBD BUFFER
001150  177564                  $TPS:   177564          ;;TTY PRINTER STATUS REG. ADDRESS
001152  177566                  $TPB:   177566          ;;TTY PRINTER BUFFER REG. ADDRESS
001154  000                     $NULL:  .BYTE   0       ;;CONTAINS NULL CHARACTER FOR FILLS
001155  002                     $FILLS: .BYTE   2       ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
001156  012                     $FILLC: .BYTE   12      ;;INSERT FILL CHARS. AFTER A "LINE FEED"
001157  000                     $TPFLG: .BYTE   0       ;;"TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
001160  000000                  $REGAD: .WORD   0       ;;CONTAINS THE ADDRESS FROM
                                                        ;;WHICH  ($REGO) WAS OBTAINED
001162  000000                  $REGO:  .WORD   0       ;;CONTAINS (($REGAD)+0)
001164  000000                  $REG1:  .WORD   0       ;;CONTAINS (($REGAD)+2)
001166  077                     $QUES:  .ASCII  /?/     ;;QUESTION MARK
001167  015                     $CRLF:  .ASCII  <15>    ;;CARRIAGE RETURN
001170  000012                  $LF:    .ASCIZ  <12>    ;;LINE FEED
                                ;;**********************************************************************
                                .SBTTL  APT MAILBOX-ETABLE

                                ;;**********************************************************************
                                .EVEN
001172                  $MAIL:                          ;;APT MAILBOX
001172  000000                 $MSGTY: .WORD   AMSGTY   ;;MESSAGE TYPE CODE
001174  000000                 $FATAL: .WORD   AFATAL   ;;FATAL ERROR NUMBER
001176  000000                 $TESTN: .WORD   ATESTN   ;;TEST NUMBER
001200  000000                 $PASS:  .WORD   APASS    ;;PASS COUNT
```

```
(2)  001202  000000          $DEVCT: .WORD   ADEVCT   ;;DEVICE COUNT
(2)  001204  000000          $UNIT:  .WORD   AUNIT    ;;I/O UNIT NUMBER
(2)  001206  000000          $MSGAD: .WORD   AMSGAD   ;;MESSAGE ADDRESS
(2)  001210  000000          $MSGLG: .WORD   AMSGLG   ;;MESSAGE LENGTH
(2)  001212                  $ETABLE:                 ;;APT ENVIRONMENT TABLE
(2)  001212     000          $ENV:   .BYTE   AENV     ;;ENVIRONMENT BYTE
(2)  001213     000          $ENVM:  .BYTE   AENVM    ;;ENVIRONMENT MODE BITS
(2)  001214  000000          $SWREG: .WORD   ASWREG   ;;APT SWITCH REGISTER
(2)  001216  000000          $USWR:  .WORD   AUSWR    ;;USER SWITCHES
(2)  001220  000000          $CPUOP: .WORD   ACPUOP   ;;CPU TYPE,OPTIONS
(2)                          ;*                       BITS 15-11=CPU TYPE
(2)                          ;*                               11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
(2)                          ;*                               11/70=06,PDQ=07,Q=10
(2)                          ;*                       BIT 10=REAL TIME CLOCK
(2)                          ;*                       BIT  9=FLOATING POINT PROCESSOR
(2)                          ;*                       BIT  8=MEMORY MANAGEMENT
(2)  001222     000          $MAMS1: .BYTE   AMAMS1   ;;HIGH ADDRESS,M.S. BYTE
(2)  001223     000          $MTYP1: .BYTE   AMTYP1   ;;MEM. TYPE,BLK#1
(2)                          ;*                       MEM.TYPE BYTE   --  (HIGH BYTE)
(2)                          ;*                               900 NSEC CORE=001
(2)                          ;*                               300 NSEC BIPOLAR=002
(2)                          ;*                               500 NSEC MOS=003
(2)  001224  000000          $MADR1: .WORD   AMADR1   ;;HIGH ADDRESS,BLK#1
(2)                          ;*                         MEM.LAST ADDR.=3 BYTES,THIS WORD AND LOW OF "TYPE" ABOVE
(2)  001226     000          $MAMS2: .BYTE   AMAMS2   ;;HIGH ADDRESS,M.S. BYTE
(2)  001227     000          $MTYP2: .BYTE   AMTYP2   ;;MEM.TYPE,BLK#2
(2)  001230  000000          $MADR2: .WORD   AMADR2   ;;MEM.LAST ADDRESS,BLK#2
(2)  001232     000          $MAMS3: .BYTE   AMAMS3   ;;HIGH ADDRESS,M.S.BYTE
(2)  001233     000          $MTYP3: .BYTE   AMTYP3   ;;MEM.TYPE,BLK#3
(2)  001234  000000          $MADR3: .WORD   AMADR3   ;;MEM.LAST ADDRESS,BLK#3
(2)  001236     000          $MAMS4: .BYTE   AMAMS4   ;;HIGH ADDRESS,M.S.BYTE
(2)  001237     000          $MTYP4: .BYTE   AMTYP4   ;;MEM.TYPE,BLK#4
(2)  001240  000000          $MADR4: .WORD   AMADR4   ;;MEM.LAST ADDRESS,BLK#4
(2)  001242  100320          $VECT1: .WORD   AVECT1   ;;INTERRUPT VECTOR#1,BUS PRIORITY#1
(2)  001244  000000          $VECT2: .WORD   AVECT2   ;;INTERRUPT VECTOR#2BUS PRIORITY#2
(2)  001246  172000          $BASE:  .WORD   ABASE    ;;BASE ADDRESS OF EQUIPMENT UNDER TEST
(2)  001250  000000          $DEVM:  .WORD   ADEVM    ;;DEVICE MAP
(2)  001252  000000          $CDW1:  .WORD   ACDW1    ;;CONTROLLER DESCRIPTION WORD#1
(2)  001254                  $ETEND:
(2)                          .MEXIT
```

```
(1)                                 .SBTTL  ERROR POINTER TABLE
(1)
(1)                                 ;*THIS TABLE CONTAINS THE INFORMATION FOR EACH ERROR THAT CAN OCCUR.
(1)                                 ;*THE INFORMATION IS OBTAINED BY USING THE INDEX NUMBER FOUND IN
(1)                                 ;*LOCATION $ITEMB. THIS NUMBER INDICATES WHICH ITEM IN THE TABLE IS PERTINENT.
(1)                                 ;*NOTE1:        IF $ITEMB IS 0 THE ONLY PERTINENT DATA IS ($ERRPC).
(1)                                 ;*NOTE2:        EACH ITEM IN THE TABLE CONTAINS 4 POINTERS EXPLAINED AS FOLLOWS:
(1)
(1)                                 ;*      EM              ;;POINTS TO THE ERROR MESSAGE
(1)                                 ;*      DH              ;;POINTS TO THE DATA HEADER
(1)                                 ;*      DT              ;;POINTS TO THE DATA
(1)                                 ;*      DF              ;;POINTS TO THE DATA FORMAT
(1)
(1)
(1)  001254                         $ERRTB:

                                    ;NO ERRORS ARE TYPED OUT

                                    .SBTTL  OPERATOR VARIABLE LOCATIONS

  001254  000020                    DELAY:  BIT4                            ;CPU DELAY FACTOR (5MS FOR 11/40 CPU)
  001256  000060                    TKBVCT: 60                              ;CONSOLE KEYBOARD VECTOR
  001260  000062                    TKBVT1: 62

                                            .SBTTL  VS-60 ADDRESSES AND INTERRUPT VECTORS

  001262  172000                    DPC:    172000                          ;DISPLAY PROGRAM COUNTER
  001264  172002                    DSR:    172002                          ;DISPLAY STATUS REGISTER
  001266  172004                    XPOS:   172004                          ;DISPLAY X AXIS REGISTER
  001270  172006                    YPOS:   172006                          ;DISPLAY Y AXIS REGISTER
  001272  172010                    DSREL:  172010                          ;DISPLAY RELOCATE REGISTER
  001274  172012                    DSR1:   172012                          ;DISPLAY EXT. STOP ADDRESS
  001276  172014                    XDOFF:  172014                          ;DISPLAY X DYNAMIC OFFSET REGISTER
  001300  172016                    YDOFF:  172016                          ;DISPLAY Y DYNAMIC OFFSET REGISTER
  001302  172020                            172020
  001304  172022                    VSCONS: 172022                          ;DISPLAY CONSOLE STATUS REGISTER
  001306  172024                            172024
  001310  172026                            172026
  001312  172030                    VSTERM: 172030                          ;DISPLAY CHARACTER TERMINATE REGISTER

  001314  000320                    DDONE:  320                             ;DISPLAY INTERRUPT VECTOR FOR STOP
  001316  000322                    DDONE1: 322

  001320  000324                    LPVCT:  324                             ;DISPLAY INTERRUPT VECTOR FOR LIGHT-PEN
  001322  000326                    LPVCT1: 326

  001324  000330                    TIMEVT: 330                             ;DISPLAY INTERRUPT VECTOR FOR TIME-OUT OR SHIFT-
  001326  000332                    TMEVT1: 332

  001330  000334                    NAMEVT: 334                             ;DISPLAY NAME MATCH VECTOR
  001332  000336                    NAMEV1: 336

                                            .SBTTL  INITIAL PROGRAM STARTUP ROUTINE
```

```
                                BEGIN:  RESET
                                .SBTTL  INITIALIZE THE COMMON TAGS
(1)                             ;;CLEAR THE COMMON TAGS ($CMTAG) AREA
(1)  001336  012706  001100             MOV     #$CMTAG,R6      ;;FIRST LOCATION TO BE CLEARED
(1)  001342  005026                     CLR     (R6)+           ;;CLEAR MEMORY LOCATION
(1)  001344  022706  001140             CMP     #SWR,R6 ;;DONE?
(1)  001350  001374                     BNE     .-6             ;;LOOP BACK IF NO
(1)  001352  012706  001100             MOV     #STACK,SP       ;;SETUP THE STACK POINTER
(1)                             ;;INITIALIZE A FEW VECTORS
(1)  001356  012737  025612  000020     MOV     #$SCOPE,@#IOTVEC ;;IOT VECTOR FOR SCOPE ROUTINE
(1)  001364  012737  000340  000022     MOV     #340,@#IOTVEC+2 ;;LEVEL 7
(1)  001372  013737  006464  006456     MOV     $ENDCT,$EOPCT   ;;SETUP END-OF-PROGRAM COUNTER
(1)  001400  012737  001400  001106     MOV     #.,$LPADR       ;;INITIALIZE THE LOOP ADDRESS FOR SCOPE
(2)                             ;;SIZE FOR A HARDWARE SWITCH REGISTER. IF NOT FOUND OR IT IS
(2)                             ;;EQUAL TO A "-1", SETUP FOR A SOFTWARE SWITCH REGISTER.
(2)  001406  013746  000004             MOV     @#ERRVEC,-(SP)  ;;SAVE ERROR VECTOR
(2)  001412  012737  001446  000004     MOV     #64$,@#ERRVEC   ;;SET UP ERROR VECTOR
(2)  001420  012737  177570  001140     MOV     #DSWR,SWR       ;;SETUP FOR A HARDWARE SWICH REGISTER
(2)  001426  012737  177570  001142     MOV     #DDISP,DISPLAY  ;;AND A HARDWARE DISPLAY REGISTER
(2)  001434  022777  177777  177476     CMP     #-1,@SWR        ;;TRY TO REFERENCE HARDWARE SWR
(2)  001442  001012                     BNE     66$             ;;BRANCH IF NO TIMEOUT TRAP OCCURRED
(2)                                                             ;;AND  THE HARDWARE SWR IS NOT = -1
(2)  001444  000403                     BR      65$             ;;BRANCH IF NO TIMEOUT
(2)  001446  012716  001454     64$:    MOV     #65$,(SP)       ;;SET UP FOR TRAP RETURN
(2)  001452  000002                     RTI
(2)  001454  012737  000176  001140 65$:MOV     #SWREG,SWR      ;;POINT TO SOFTWARE SWR
(2)  001462  012737  000174  001142     MOV     #DISPREG,DISPLAY
(2)  001470  012637  000004     66$:    MOV     (SP)+,@#ERRVEC  ;;RESTORE ERROR VECTOR
(1)
(2)  001474  005037  001200             CLR     $PASS           ;;CLEAR PASS COUNT
(2)  001500  132737  000200  001213     BITB    #APTSIZE,$ENVM  ;;TEST USER SIZE UNDER APT
(2)  001506  001403                     BEQ     67$             ;;YES,USE NON-APT SWITCH
(2)  001510  012737  001214  001140     MOV     #$SWREG,SWR     ;;NO,USE APT SWITCH REGISTER
(2)  001516                     67$:
 70  001516  012700  001262     RESTAT: MOV     #DPC,R0                 ;GET POINTER
 71  001522  013701  001246             MOV     $BASE,R1                ;GET SUPPLIED ADDRESS
 72  001526  010120             1$:     MOV     R1,(R0)+                ;UPDATE
 73  001530  062701  000002             ADD     #2,R1                   ;THE
 74  001534  022700  001314             CMP     #DDONE,R0               ;ADDRESSES
 75  001540  001372                     BNE     1$                      ;UNTIL DONE
 76  001542  012700  001314             MOV     #DDONE,R0               ;GET POINTER
 77  001546  013701  001242             MOV     $VECT1,R1               ;GET SUPPLIED VECTOR
 78  001552  042701  160000             BIC     #160000,R1              ;CLEAR PSW BITS
 79  001556  010120             2$:     MOV     R1,(R0)+                ;UPDATE
 80  001560  062701  000002             ADD     #2,R1                   ;THE VECTORS
 81  001564  022700  001334             CMP     #DDONE+20,R0
 82  001570  001372                     BNE     2$
 83  001572  005037  007622             CLR     SWITCH                  ;HOUSEKEEP
 84  001576  005037  007506             CLR     HOLD                    ;
 85  001602  005004                     CLR     R4                      ;
 86  001604  005037  007510             CLR     TSAVE                   ;
 87  001610  013777  001260  177440     MOV     TKBVT1,@TKBVCT          ;RESET KRB VECTOR
 88  001616  005077  177436             CLR     @TKBVT1
 89  001622  005037  002230             CLR     KRBD
```

```
                                                TSTB    @SWR                    ;TEST FOR "KRB" CONTROL
                                                BEQ     3$                      ;BR IF NOT
                                                COM     KRBD                    ;SET "KRB" CONTROL
                                                MOV     #RETB,@TKBVCT           ;SET UP "KRB" INT
                                                MOV     #340,@TKBVT1
                                        3$:     JSR     PC,FIXVCT               ;LOAD BUS VECTORS
                                                JMP     TST1                    ;START TESTING
                                        FIXVCT: MOV     #MESGAA,@DDONE          ;SET UP VS-60 DONE VECTOR
                                                MOVB    $VECT1+1,R0             ;GET BR LEVEL
                                                BIC     #177400,R0              ;MASK OFF OTHER BITS
                                                MOV     R0,@DDONE1
                                                MOV     LPVCT1,@LPVCT           ;RESET LIGHT-PEN VECTOR
                                                CLR     @LPVCT1
                                                MOV     TMEVT1,@TIMEVT          ;RESET TIME-OUT/SHIFT OUT VECTOR
                                                CLR     @TMEVT1
                                                MOV     NAMEV1,@NAMEVT          ;RESET NAME MATCH VECTOR
                                                CLR     @NAMEV1
                                                MOV     #30060,DLT14A           ;INIT X READOUT VALUE FOR CONSOLE #0
                                                MOV     #30060,DLT14A+2
                                                MOV     #30060,DLT14B           ;INIT Y READOUT
                                                MOV     #30060,DLT14B+2
                                                MOV     #30060,DLT14C           ;RESET READOUT VALUE FOR CONSLE #1
                                                MOV     #30060,DLT14C+2
                                                MOV     #30060,DLT14D
                                                MOV     #30060,DLT14D+2
                                                RTS     PC                      ;EXIT
                                                .SBTTL  KEYBOARD SERVICE ROUTINE
                                        RETB:   MOVB    @$TKB,TSAVE             ;READ THE CHARACTER
                                                BIC     #177600,TSAVE           ;MASK TO 7 BITS
                                                CMP     #3,TSAVE                ;TEST FOR "CTRL C"
                                                BNE     7$                      ;BR IF NOT
                                                JMP     BEGIN
                                        7$:     CMP     #15,TSAVE               ;TEST FOR "CR"
                                                BEQ     5$                      ;BR IF
                                                CLR     SWITCH                  ;CLEAR "SWITCH"
                                                SUB     #101,TSAVE              ;MAKE 0-77
                                        1$:     BMI     4$                      ;<A
                                                CMP     #32,TSAVE
                                                BMI     3$                      ;>Z
                                                MOV     TSAVE,R4
                                                MOVB    R4,$TSTNM               ;LOAD TEST #
                                                MOV     #STACK,SP               ;RELOAD STACK
                                                ASL     R4
                                                CLR     SWITCH
                                                CLR     HOLD
                                                RESET
                                                JSR     PC,FIXVCT               ;RESET DISPLAY VECTORS
                                                TST     DISPTC(R4)              ;TEST IF VALID
                                                BNE     2$
                                                CLR     R4
                                        2$:     JMP     @DISPTC(R4)             ;EXIT TO THAT TEST SELECTED
                                        3$:     CMP     #76,TSAVE
                                                BNE     6$
                                                MOV     #-1,HOLD                ; RUBOUT
```

```
                                        RTI                             ;EXIT
002200  000002
002202  005037  007506          4$:     CLR     HOLD
002206  000002                          RTI
002210  012737  177777  007622  5$:     MOV     #-1,SWITCH              ;SET "STOP MOTION" FLAG
002216  000002                          RTI
002220  162737  000040  007510  6$:     SUB     #40,@SAVE               ;CONVERT LC TO UC
002226  000724                          BR      1$
002230  000000                  KRBD:   0

                                        .SBTTL  VS-60 INSTRUCTION SET

                100000                  CHAR=100000                     ;DISPLAY IN CHARACTER MODE
                104000                  SHORTV=104000                   ;               SHORT VECTOR
                110000                  LONGV=110000                    ;               LONG VECTOR MODE
                114000                  POINT=114000                    ;               POINT MODE
                120000                  GRAPHX=120000                   ;               GRAPHPLOT X MODE
                124000                  GRAPHY=124000                   ;               GRAPHPLOT Y MODE
                120000                  BASICV=GRAPHX                   ;               BASIC VECTOR MODE
                130000                  RELATP=130000                   ;               RELATIVE POINT MODE
                134000                  BASICS=RELATP!4000              ;               BASIC SHORT VECTOR MODE
                144000                  ABSVCT=144000                   ;               ABSOLUTE VECTOR MODE

                010000                  OFFST0=10000                    ;               ENABLE OFFSET OF 0
                012000                  OFFST1=12000                    ;               ENABLE OFFSET OF 1
                014000                  OFFST2=14000                    ;               ENABLE OFFSET OF 2
                016000                  OFFST3=16000                    ;               ENABLE OFFSET OF 3

                002000                  INT0=2000                       ;ENABLE INTENSITY LEVEL 0
                002200                  INT1=2200                       ;                       1
                002400                  INT2=2400                       ;                       2
                002600                  INT3=2600                       ;                       3
                003000                  INT4=3000                       ;                       4
                003200                  INT5=3200                       ;                       5
                003400                  INT6=3400                       ;                       6
                003600                  INT7=3600                       ;LEVEL 7

                000100                  LPOFF=100                       ;DISABLE LIGHT-PEN INTERRUPT
                000140                  LPON=140
                000020                  BLKOFF=20                       ;DISABLE BLINK
                000030                  BLKON=30                        ;ENABLE BLINK

                000004                  LINE0=4                         ;ENABLE LINE TYPE 0
                000005                  LINE1=5                         ;ENABLE LINE TYPE 1
                000006                  LINE2=6                         ;ENABLE LINE TYPE 2
                000007                  LINE3=7                         ;ENABLE LINE TYPE 3

                002000                  PATH0=2000                      ;DIRECTION 0
                006000                  PATH1=6000                      ;DIRECTION 1
                012000                  PATH2=12000                     ;DIRECTION 2
                016000                  PATH3=16000                     ;DIRECTION 3
                022000                  PATH4=22000                     ;          4
                026000                  PATH5=26000                     ;          5
                032000                  PATH6=32000                     ;          6
                036000                  PATH7=36000                     ;          7
```

```
                        
160000                  DJMP=160000             ;DISPLAY ABSOLUTE JUMP
161000                  DJMPR=DJMP!BIT9         ;DISPLAY RELATIVE JUMP
162000                  DJSR=DJMP!BIT10         ;DISPLAY JSR ABSOLUTE
163000                  DJSRR=DJSR!BIT9         ;DISPLAY JSR RELATIVE
                        
164000                  DNOP=164000
166000                  DPOP=DNOP!BIT10         ;POP AND RESTORE
165000                  DPOPNR=DNOP!BIT9        ;POP AND NO RESTORE
164000                  CONSL0=DNOP             ;CONSOLE 0
164400                  CONSL1=DNOP!BIT8        ;CONSOLE 1
                        
170000                  STATSA=170000
173400                  DSTOP=173400
170002                  DMENU0=STATSA!BIT1      ;DISABLE MENU
170003                  DMENU1=DMENU0!BIT0
                        
000200                  LPLITE=200
000300                  LPDARK=300
000040                  ITAL0=40                ;DISABLE ITALIC CHARACTERS
000060                  ITAL1=60
000004                  SYNC30=4                ;ENABLE SYNC OF 30 CPS
000010                  SYNC40=10               ;ENABLE SYNC OF 40 CPS
                        
174000                  STATSB=174000
                        
000100                  INCR=100                ;ENABLE "GRAPHPLOT INCREMENT REG. CHANGE"
                        
154000                  STATSC=154000
155000                  CHRRT0=STATSC!BIT9      ;DISABLE CHAR ROTATE
155400                  CHRRT1=CHRRT0!BIT8
                        
154200                  CHARS0=STATSC!BIT7      ;LOAD CHARACTER SCALE TO 1/2
154240                  CHARS1=CHARS0!BIT5      ;                        1
154300                  CHARS2=CHARS0!BIT6      ;                        1 1/2
154340                  CHARS3=CHARS0!BIT6!BIT5 ;                        2
                        
154020                  VCTR00=STATSC!BIT4      ;LOAD VECTOR SCALE REGISTER
                        
176000                  STATE=STATSB!BIT10
                        
176002                  STRNG0=STATE!BIT1       ;DISABLE CHARACTER STRING TERMINATE
176003                  STRNG1=STRNG0!BIT0
                        
176040                  EDGE0=STATE!BIT5        ;DISABLE EDGE INTERRUPT
176060                  EDGE1=EDGE0!BIT4
150000                  DNAME=150000            ;LOAD DISPLAY NAME REGISTER
                        
                ;MORE EQUATES
                        
040000                  INTX=BIT14              ;INTENSIFY
000177                  MAXMUX=177              ;MAX. MENU  X WIDTH
001777                  MAXX=1777               ;MAX. X AXIS LENGTH
001777                  MAXY=1777               ;MAX. Y AXIS LENGTH
```

```
                000777                  HALFX=MAXX/2                    ;HALF OF MAXIUM LENGTH
                020000                  MINUSX=20000                    ;NEGATIVE SIGN BIT
                020000                  MINUSY=20000                    ;NEGATIVE SIGN BIT
                000100                  MINSUY=100                      ;NEGATIVE SIGN BIT (SHORT VECTOR MODE)
                000021                  SUPON=21                        ;SUPER-SCRIPT ENABLE
                000023                  SUPOFF=23                       ;SUPER-SCRIPT DISABLE
                000022                  SUBON=22                        ;SUB-SCRIPT ENABLE
                000024                  SUBOFF=24                       ;SUB-SCRIPT DISABLE


                                        .SBTTL
                                        .SBTTL  TEST    LETTER          DESCRIPTION
                                        .SBTTL  ----    ------          -----------
                                        .SBTTL

                                ;;**********************************************************************
                                ;*TEST 1         A       DIRECTORY FRAME
                                ;;**********************************************************************
002232  000004                  TST1:   SCOPE

002234  004537  007520                  JSR     R5,MESG                 ;EXIT TO DISPLAY A FRAME
002240  001000                          1000
002242  010006                          FRMEO                           ;USING THE DIR. FRAME


                                ;;**********************************************************************
                                ;*TEST 2         B       ASTIGMATISM AND SETTLING TIME
                                ;;**********************************************************************
002244  000004                  TST2:   SCOPE
002246  004537  007520                  JSR     R5,MESG                 ;DISPLAY DATA LOCATED IN "BUFFER"
002252  020000                          20000
002254  012044                          TABB
```

```
                                ;;*****************************************************************
                                ;*TEST 3        C       SHORT TERM DRIFT
                                ;;*****************************************************************
002256  000004                  TST3:   SCOPE
002260  012737  000100  007514          MOV     #BIT6,TEMPA             ;LOAD EXECUTION COUNT
002266  012700  007704          1$:     MOV     #TABA,R0                ;LOAD TABLE POINTER

002272  012037  017536          2$:     MOV     (R0)+,STDRA             ;LOAD X POSITION DATA
002276  012037  017540                  MOV     (R0)+,STDRB             ;LOAD Y POSITION DATA

002302  100441                          BMI     3$
002304  004537  007520                  JSR     R5,MESG                 ;LOAD X+Y POSITION-DO NOT DISPLAY
002310  000001                          1
002312  017532                          STDPIC

002314  052737  040000  017536          BIS     #INTX,STDRA             ;LOAD INTENSIFY ENABLE

002322  004537  007520                  JSR     R5,MESG                 ;DISPLAY DATA
002326  000001                          1
002330  017532                          STDPIC
002332  004537  002416                  JSR     R5,SECDLY               ;DELAY FOR 5 MSEC
002336  000005                          5
002340  004537  007520                  JSR     R5,MESG                 ;DISPLAY POINT AGAIN
002344  000001                          1
002346  017532                          STDPIC
002350  004537  002416                  JSR     R5,SECDLY               ;DELAY FOR 5 MSEC
002354  000005                          5

002356  004537  007520                  JSR     R5,MESG                 ;DISPLAY POINT AGAIN
002362  000001                          1
002364  017532                          STDPIC
002366  004537  002416                  JSR     R5,SECDLY               ;DELAY FOR 5 MSEC
002372  000005                          5

002374  004537  007520                  JSR     R5,MESG                 ;DISPLAY POINT AGAIN
002400  000001                          1
002402  017532                          STDPIC
002404  000732                          BR      2$

002406  005337  007514          3$:     DEC     TEMPA                   ;FINISHED EXECUTION?
002412  001325                          BNE     1$
002414  000416                          BR      TST4            ;;BR OVER SUBROUTINE

002416  012537  002450          SECDLY: MOV     (R5)+,11$               ;LOAD TOTAL DELAY COUNT
002422  013737  001254  002446  2$:     MOV     DELAY,10$               ;LOAD 1 MS.
002430  005337  002446          1$:     DEC     10$                     ;DELAY
002434  001375                          BNE     1$
002436  005337  002450                  DEC     11$                     ;DEC MSEC COUNT
002442  100367                          BPL     2$
002444  000205                          RTS     R5                      ;EXIT
002446  000000                  10$:    0
002450  000000                  11$:    0
```

```
                        ;;*****************************************************************
                        ;*TEST 4        D       MINOR AXIS GAIN, OFFSET AND PHASE ADJUSTMENT
                        ;;*****************************************************************
002452  000004         TST4:   SCOPE
002454  012737  000010  007514          MOV     #BIT3,TEMPA             ;LOAD EXECUTION COUNT
002462  004537  007520  1$:     JSR     R5,MESG                 ;DISPLAY SUB-PICTURE
002466  001000                  1000
002470  012306                  FRME2
002472  005337  007514          DEC     TEMPA                   ;FINISHED EXECUTION ?
002476  001371                  BNE     1$                      ;BR IF NOT

                        ;;*****************************************************************
                        ;*TEST 5        E       MAJOR AXIS OFFSET AND VECTOR START POINT ADJUSTMENT
                        ;;*****************************************************************
002500  000004         TST5:   SCOPE
002502  012737  001000  007514          MOV     #BIT9,TEMPA
002510  004537  007520  1$:     JSR     R5,MESG                 ;DISPLAY OFFSET SUB-PICTURE
002514  000010                  10
002516  012306                  FRME2
002520  004537  007520          JSR     R5,MESG                 ;DISPLAY VECTOR START SUB-PICTURE
002524  000010                  10
002526  020174                  VSTRT
002530  005337  007514          DEC     TEMPA                   ;FINISHED EXECUTION LOOP?
002534  001365                  BNE     1$                      ;BR IF NOT DONE.

                        ;;*****************************************************************
                        ;*TEST 6        F       VECTOR LENGTH GAIN, CONVERGENCE AND VECTOR LINEARITY
                        ;;*****************************************************************
002536  000004         TST6:   SCOPE
                        ;GENERATE THE SCREEN PICTURE BUFFER FIRST
002540  012700  026070          MOV     #BUFFER,R0              ;LOAD DISPLAY PICTURE POINTER
002544  012720  164700          MOV     #CONSL1!BIT7!BIT6,(R0)+         ;ENABLE CONSOLE #1
002550  012701  012372          MOV     #PICST0,R1              ;LOAD "BOX POINTER"
002554  012120          1$:     MOV     (R1)+,(R0)+             ;GET DATA INTO BUFFER
002556  022701  012422          CMP     #PICST1,R1              ;TEST FOR END
002562  001374                  BNE     1$                      ;BR IF NOT

002564  012737  001777  017552          MOV     #MAXX,PICVTA            ;LOAD STARTING X POSITION
002572  012737  000040  017554          MOV     #40,PICVTB              ;LOAD STARTING Y POSITION
002600  012737  060040  017560          MOV     #INTX!MINUSX+40,PICVTC  ;LOAD INTENFIFY, MINUS DIR AND VALUE

002606  012737  000037  007514          MOV     #37,TEMPA               ;LOAD A COUNTER
002614  012701  017550  3$:     MOV     #PICVTL,R1              ;LOAD SUB-PICTURE POINTER
002620  012120          2$:     MOV     (R1)+,(R0)+             ;LOAD DATA
002622  022701  017564          CMP     #PICVTE,R1              ;TEST FOR END
002626  001374                  BNE     2$                      ;BR IF NOT
002630  005337  007514          DEC     TEMPA
002634  001407                  BEQ     4$                      ;BR IF DONE THIS SIDE
002636  062737  000040  017554          ADD     #40,PICVTB              ;ADJUST STARTING Y POSITION
002644  062737  000040  017560          ADD     #40,PICVTC              ;ADJUST VECTOR LENGTH
002652  000760                  BR      3$
002654  012737  000000  017552  4$:     MOV     #0,PICVTA               ;LOAD STARTING X POSITION
```

```
002662 012737 000040 017554          MOV     #40,PICVTB                    ;LOAD STARTING Y POSITION
002670 012737 041740 017560          MOV     #INTX+1740,PICVTC             ;LOAD INTENSIFY AND DELTA VALUE

002676 012737 000037 007514          MOV     #37,TEMPA                     ;LOAD A COUNTER
002704 012701 017550         5$:     MOV     #PICVTL,R1                    ;LOAD SUB-PICTURE POINTER
002710 012120                6$:     MOV     (R1)+,(R0)+                   ;LOAD DATA
002712 022701 017564                 CMP     #PICVTE,R1                    ;TEST FOR END
002716 001374                        BNE     6$                            ;BR IF NOT
002720 005337 007514                 DEC     TEMPA                         ;TEST IF DONE
002724 001407                        BEQ     7$                            ;BR IF SUB-PICTURE
002726 062737 000040 017554          ADD     #40,PICVTB                    ;ADJUST STARTING Y POSITION
002734 162737 000040 017560          SUB     #40,PICVTC                    ;ADJUST VECTOR LENGTH
002742 000760                        BR      5$                            ;BR BACK
002744 012720 114000         7$:     MOV     #POINT,(R0)+                  ;LOAD POINT INST
002750 012701 040000                 MOV     #INTX,R1                      ;LOAD STARTING X POSITION
002754 012702 001777                 MOV     #MAXY,R2                      ;LOAD STARTING Y POSITION
002760 010120                8$:     MOV     R1,(R0)+                      ;LOAD X POSITION
002762 010220                        MOV     R2,(R0)+                      ;LOAD Y POSITION
002764 062701 000040                 ADD     #40,R1                        ;ADJUST X
002770 162702 000040                 SUB     #40,R2                        ;ADJUST Y
002774 100371                        BPL     8$                            ;BR IF NOT DONE
002776 012720 114000                 MOV     #POINT,(R0)+
003002 012720 000000                 MOV     #0,(R0)+
003006 012720 001777                 MOV     #MAXY,(R0)+                   ;LOAD POINT IN UPPER LEFT CORN
003012 012720 110000                 MOV     #LONGV,(R0)+                  ;LOAD DECENDING DIAG. LINE
003016 012720 041777                 MOV     #INTX!MAXX,(R0)+
003022 012720 021777                 MOV     #MINUSX!MAXY,(R0)+
                             ;DRAW BASIC VECTOR SECTION
003026 012720 114000                 MOV     #POINT,(R0)+
003032 012720 001000                 MOV     #1000,(R0)+
003036 012720 001000                 MOV     #1000,(R0)+
003042 012720 120000                 MOV     #BASICV,(R0)+                 ;LOAD BASIC VECTOR
003046 012720 042777                 MOV     #INTX!PATH0!HALFX,(R0)+               ;DISPLAY BASIC VECTOR
003052 012720 062777                 MOV     #INTX!PATH4!HALFX,(R0)+
003056 012720 052777                 MOV     #INTX!PATH2!HALFX,(R0)+
003062 012720 072777                 MOV     #INTX!PATH6!HALFX,(R0)+
003066 012720 062777                 MOV     #INTX!PATH4!HALFX,(R0)+
003072 012720 042777                 MOV     #INTX!PATH0!HALFX,(R0)+
003076 012720 072777                 MOV     #INTX!PATH6!HALFX,(R0)+
003102 012720 052777                 MOV     #INTX!PATH2!HALFX,(R0)+
003106 012720 173400                 MOV     #DSTOP,(R0)+
003112 012720 160000                 MOV     #DJMP,(R0)+
003116 012720 026070                 MOV     #BUFFER,(R0)+


                             ;THE PICTURE HAS NOW BEEN COMPLETED
003122 012737 003130 001106          MOV     #20$,$LPADR                   ;RESET LOOP ADDRESS
003130 004537 007520         20$:    JSR     R5,MESG                       ;EXIT TO DISPLAY ROUTINE
003134 002000                        2000
003136 026070                        BUFFER                                ;USING BUFFER STORAGE
```

```
                                ;;*****************************************************************
                                ;*TEST 7         G       PINCUSHION FRAME
                                ;;*****************************************************************
003140  000004                  TST7:   SCOPE
003142  012700  026070                  MOV     #BUFFER,R0                      ;LOAD START ADDRESS
003146  012720  164700                  MOV     #CONSL1!BIT7!BIT6,(R0)+         ;ENABLE CONSOL #1
003152  004737  003322                  JSR     PC,3$                           ;LOAD 0,0 ORGIN
003156  012701  000020                  MOV     #20,R1                          ;SETUP COUNT
003162  012720  040000          1$:     MOV     #INTX,(R0)+                     ;LOAD INT LINE
003166  012720  001777                  MOV     #MAXY,(R0)+                     ; MAX Y
003172  012720  000100                  MOV     #100,(R0)+                      ;LOAD DELTA X
003176  012720  021777                  MOV     #MINUSX+MAXY,(R0)+              ;LOAD - MAX Y
003202  005301                          DEC     R1                              ;FINISHED ?
003204  001366                          BNE     1$                              ;BR IF NOT
003206  012720  020001                  MOV     #MINUSX+1,(R0)+                 ;GO BACK 1 UNIT
003212  012720  000000                  MOV     #0,(R0)+
003216  012720  040000                  MOV     #INTX,(R0)+
003222  012720  001777                  MOV     #MAXY,(R0)+                     ;PLOT LAST LINE
003226  004737  003322                  JSR     PC,3$                           ;SET ORGIN
003232  012701  000020                  MOV     #20,R1                          ;SETUP COUNT
003236  012720  041777          2$:     MOV     #INTX+MAXX,(R0)+                ;LOAD DELTA X MAX
003242  012720  000000                  MOV     #0,(R0)+                        ;LOAD DELTA Y = 0
003246  012720  021777                  MOV     #MINUSX+MAXX,(R0)+              ;RETRACE
003252  012720  000100                  MOV     #100,(R0)+                      ;LOAD DELTA Y OF 100
003256  005301                          DEC     R1                              ;FINISHED ?
003260  001366                          BNE     2$                              ;BR IF NOT
003262  012720  000000                  MOV     #0,(R0)+
003266  012720  020001                  MOV     #MINUSX+1,(R0)+
003272  012720  041777                  MOV     #INTX+MAXX,(R0)+                ;PLOT LAST LINE
003276  012720  000000                  MOV     #0,(R0)+
003302  012720  173400                  MOV     #DSTOP,(R0)+                    ;LOAD STOP
003306  012720  160000                  MOV     #DJMP,(R0)+                     ;LOAD JUMP
003312  012710  026070                  MOV     #BUFFER,(R0)
003316  000137  003344                  JMP     4$

003322  012720  117000          3$:     MOV     #POINT!INT4,(R0)+               ;LOAD POINT
003326  012720  000000                  MOV     #0,(R0)+                        ; AT X
003332  012720  000000                  MOV     #0,(R0)+                        ; AT Y
003336  012720  110000                  MOV     #LONGV,(R0)+                    ;LONG VECTOR
003342  000207                          RTS     PC                              ;EXIT

003344  004537  007520          4$:     JSR     R5,MESG                         ;EXIT TO DISPLAY FRAME
003350  004000                          4000
003352  026070                          BUFFER                                  ;USING THE CROSS HATCH PATTERN
```

```
                                ;;*********************************************************************
                                ;*TEST 10        H       OCTAGONS AND CIRCLES
                                ;;*********************************************************************
003354  000004                  TST10:  SCOPE
003356  004537  007520                  JSR     R5,MESG                 ;DISPLAY TEST
003362  006000                          6000
003364  013336                          FRME3                           ;FRAME # 3

                                ;;*********************************************************************
                                ;*TEST 11        I       SCISSORING AND VECTOR SCALING
                                ;;*********************************************************************
003366  000004                  TST11:  SCOPE
003370  012737  000400  007516          MOV     #BIT8,TEMPB             ;LOAD EXECUTION COUNTER
003376  012737  154037  020134  1$:     MOV     #VCTR00!17,PICSCA       ;RELOAD VECTOR SCALE LENGTH TO 17
003404  012737  000020  007514          MOV     #20,TEMPA               ;LOAD SCALE COUNTER

003412  004537  007520          2$:     JSR     R5,MESG                 ;EXIT TO DISPLAY ROUTINE
003416  000001                          1
003420  017572                          PICSCS                          ;USING PRESET PICTURE DATA

003422  005337  020134                  DEC     PICSCA                  ;REDUCE VECTOR SCALE
003426  005337  007514                  DEC     TEMPA                   ;FINISHED ALL SCALES?
003432  001367                          BNE     2$                      ;BR IF NOT

003434  005337  007516                  DEC     TEMPB                   ;FINISHED EXECUTION COUNT
003440  001356                          BNE     1$                      ;BR IF NOT
```

```
                                        ;;*****************************************************************
                                        ;*TEST 12       J       OFFSET X AND OFFSET Y POSITION
                                        ;;*****************************************************************
                                        ;
(2)  003442  000004                     TST12:  SCOPE
                                        ;DISPLAY A SQUARE IN THE CENTER SCREEN, THEN
                                        ;MOVE THE BOX TO THE RIGHT
003444  012737  010000  015036                  MOV     #BIT12,OFFT1            ;LOAD BASIC X OFFSET VALUE
003452  012737  010000  015040                  MOV     #BIT12,OFFT2            ;LOAD BASIC Y OFFSET VALUE
003460  004537  007520          1$:     JSR     R5,MESG                         ;DISPLAY THAT FRAME
003464  000100                          100
003466  015034                          OFFTST
003470  005737  007622                  TST     SWITCH                          ;TEST IF HOLD HERE
003474  001371                          BNE     1$                              ;BR IF YES
003476  062737  000001  015036          ADD     #1,OFFT1                        ;UPDATE THE X OFFSET
003504  022737  011400  015036          CMP     #BIT12!1400,OFFT1               ;TEST IF MORE TO MOVE
003512  001362                          BNE     1$                              ;BR IF NOT
                                        ;MOVE THE BOX TO THE LEFT
003514  012737  030000  015036          MOV     #BIT12!MINUSX,OFFT1             ;LOAD THE BASIC X OFFSET
003522  012737  010000  015040          MOV     #BIT12,OFFT2                    ;LOAD THE BASIC Y OFFSET
003530  004537  007520          2$:     JSR     R5,MESG                         ;DISPLAY THE FRAME
003534  000100                          100
003536  015034                          OFFTST
003540  005737  007622                  TST     SWITCH                          ;TEST IF HOLD HERE
003544  001371                          BNE     2$                              ;BR IF HOLD
003546  062737  000001  015036          ADD     #1,OFFT1                        ;UPDATE THE X OFFSET
003554  022737  031400  015036          CMP     #BIT12!MINUSX!1400,OFFT1                ;TEST IF MORE
003562  001362                          BNE     2$                              ;BR IF NOT
                                        ;MOVE THE BOX UP
003564  012737  010000  015036          MOV     #BIT12,OFFT1                    ;LOAD BASIC X OFFSET
003572  012737  010000  015040          MOV     #BIT12,OFFT2                    ;LOAD BASIC Y OFFSET
003600  004537  007520          3$:     JSR     R5,MESG                         ;DISPLAY THAT FRAME
003604  000100                          100
003606  015034                          OFFTST
003610  005737  007622                  TST     SWITCH                          ;TEST IF HOLD HERE
003614  001371                          BNE     3$                              ;BR IF YES
003616  062737  000001  015040          ADD     #1,OFFT2                        ;UPDATE Y OFFSET
003624  022737  011400  015040          CMP     #BIT12!1400,OFFT2               ;TEST IF MORE
003632  001362                          BNE     3$                              ;BR IF NOT
                                        ;MOVE THE BOX DOWN
003634  012737  030000  015040          MOV     #BIT12!MINUSY,OFFT2             ;LOAD THE BASIC Y OFFSET
003642  012737  010000  015036          MOV     #BIT12,OFFT1                    ;LOAD THE BASIC X OFFSET
003650  004537  007520          4$:     JSR     R5,MESG                         ;DISPLAY THAT FRAME
003654  000100                          100
003656  015034                          OFFTST
003660  005737  007622                  TST     SWITCH                          ;TEST IF HOLD HERE
003664  001371                          BNE     4$                              ;BR IF YES
003666  062737  000001  015040          ADD     #1,OFFT2                        ;UPDATE Y OFFSET
003674  022737  031400  015040          CMP     #BIT12!MINUSX!1400,OFFT2        ;TEST IF MORE
003702  001362                          BNE     4$                              ;BR IF NOT
```

```
                                ;;*****************************************************************
                                ;*TEST 13        K       CHARACTER SCALE FRAME
                                ;;*****************************************************************
003704  000004                  TST13:  SCOPE
003706  012737  000200  007514          MOV     #BIT7,TEMPA             ;LOAD EXECUTION COUNTER

003714  004537  007520          1$:     JSR     R5,MESG                 ;DISPLAY SUBPICTURE DATA
003720  000060                          60
003722  020502                          CHAQU

003724  005337  007514                  DEC     TEMPA                   ;FINISHED EXECUTION?
003730  001371                          BNE     1$                      ;BR IF NOT


                                ;;*****************************************************************
                                ;*TEST 14        L       CHARACTER QUALITY AND CHARACTER ROTATE IN THE MENU
                                ;;*****************************************************************
003732  000004                  TST14:  SCOPE
003734  012700  026070                  MOV     #BUFFER,R0              ;LOAD DESTINATION POINTER
003740  012720  164700                  MOV     #CONSL1!BIT7!BIT6,(R0)+ ;ENABLE CONSOLE #1
003744  012720  114000                  MOV     #POINT,(R0)+            ;LOAD "DPOINT"
003750  005020                          CLR     (R0)+                   ;LOAD X AXIS
003752  012720  001763                  MOV     #MAXY-14,(R0)+          ;LOAD Y
003756  012701  000015                  MOV     #15,R1                  ;LOAD COUNT
003762  012720  162000          1$:     MOV     #DJSR,(R0)+             ;LOAD "DJSR" TO BUFFER SPACE
003766  012720  021240                  MOV     #CHARQA,(R0)+           ;LOAD TARGET ADDRESS
003772  005301                          DEC     R1                      ;FINISHED ?
003774  001372                          BNE     1$                      ;BR UNTIL DONE
003776  012720  160000                  MOV     #DJMP,(R0)+
004002  012720  021172                  MOV     #ROTCHR,(R0)+           ;RETURN ADDRESS TO THE ROTATED CHAR SUB-PIC

004006  004537  007520                  JSR     R5,MESG                 ;EXECUTE DISPLAY FILE
004012  000200                          200
004014  026070                          BUFFER                          ;STARTING AT "BUFFER" ADDRESS
```

```
                        ;;*****************************************************************
                        ;*TEST 15       M       CHARACTER SET, SUPERSCRIPT, SUBSCRIPT AND ITALICS
                        ;;*****************************************************************
004016  000004          TST15:  SCOPE
                        ;SET UP THE BUFFER FOR THIS TEST
004020  012700  026070          MOV     #BUFFER,R0
004024  012720  155000          MOV     #CHRRTO,(R0)+           ;DISABLE CHAR. ROTATE
004030  012720  164700          MOV     #CONSL1!BIT7!BIT6,(R0)+ ;ENABLE CONSOLE #1
004034  012720  114000          MOV     #POINT,(R0)+            ;LOAD POINT MPDE
004040  005020                  CLR     (R0)+
004042  012720  001700          MOV     #MAXY-77,(R0)+
004046  012720  100000          MOV     #CHAR,(R0)+
004052  012737  000100  004252  MOV     #100,STCHAR             ;LOAD INITIAL CHAR.
004060  004737  004210          JSR     PC,LOADBF
004064  012737  000140  004252  MOV     #140,STCHAR             ;LOAD INITIAL LC CHAR
004072  004737  004210          JSR     PC,LOADBF               ;LOAD LINE
004076  012737  000040  004252  MOV     #40,STCHAR              ;LOAD NUMBERS AND PUNCT
004104  004737  004210          JSR     PC,LOADBF               ;LOAD LINE
004110  012720  170040          MOV     #STATSA!ITAL0,(R0)+     ;LOAD NORMAL FONT
004114  004737  004154          JSR     PC,LOADSP               ;LOAD SPECIAL CHARS
004120  004737  004314          JSR     PC,SPACE                ;INSERT SPACES
004124  012720  170060          MOV     #STATSA!ITAL1,(R0)+     ;LOAD ITALICS FONT
004130  004737  004154          JSR     PC,LOADSP               ;LOAD SPIECAL
004134  012720  173400          MOV     #DSTOP,(R0)+            ;LOAD DSTOP
004140  012720  160000          MOV     #DJMP,(R0)+
004144  012720  026070          MOV     #BUFFER,(R0)+
004150  000137  004332          JMP     FILE4A

004154  112720  000016  LOADSP: MOVB    #16,(R0)+               ;LOAD "SHIFT-OUT" CHARACTER
004160  005002                  CLR     R2                      ;SET INITIAL SHIFT OUT CHAR
004162  110220          1$:     MOVB    R2,(R0)+                ;LOAD CHAR
004164  005202          2$:     INC     R2
004166  022702  000017          CMP     #17,R2                  ;TEST FOR "SHIFT-IN" (SI)
004172  001774                  BEQ     2$                      ;BR IF SI "17"
004174  022702  000040          CMP     #40,R2                  ;FINISHED ?
004200  001370                  BNE     1$                      ;BR IF NOT
004202  012720  020017          MOV     #20017,(R0)+            ;LOAD SHIFT-IN SPACE
004206  000207                  RTS     PC                      ;EXIT

004210  012720  170040  LOADBF: MOV     #STATSA!ITAL0,(R0)+     ;LOAD NORMAL FONT
004214  013702  004252          MOV     STCHAR,R2               ;GET STARTING CHAR
004220  004737  004276          JSR     PC,FILLIT               ;LOAD THE CHARACTERS
004224  004737  004314          JSR     PC,SPACE                ;INSERT SPACES
004230  012720  170060          MOV     #STATSA!ITAL1,(R0)+     ;LOAD ITALICS FONT
004234  013702  004252          MOV     STCHAR,R2               ;GET STARTING CHARACTER
004240  004737  004276          JSR     PC,FILLIT               ;LOAD THE CHARACTERS
004244  004737  004254          JSR     PC,ACRLF                        ;INSERT CR-LF
004250  000207                  RTS     PC                      ;EXIT

004252  000000  STCHAR: 0
```

```
                                ;LOAD CR-LF'S TO VERTICALLY SPACE THE STRINGS

004254  112720  000015        ACRLF:  MOVB    #15,(R0)+
004260  112720  000012                MOVB    #12,(R0)+
004264  112720  000012                MOVB    #12,(R0)+
004270  112720  000012                MOVB    #12,(R0)+
004274  000207                        RTS     PC                      ;EXIT

                                ;FILL IN WITH AN INCREMENTING CHARACTERS

004276  012703  000040        FILLIT: MOV     #40,R3
004302  110220                1$:     MOVB    R2,(R0)+
004304  005202                        INC     R2
004306  005303                        DEC     R3
004310  001374                        BNE     1$
004312  000207                        RTS     PC

                                ;LOAD "SPACE" CHAR TO SEPERATE CHAR STRINGS

004314  012703  000010        SPACE:  MOV     #10,R3
004320  112720  000040        1$:     MOVB    #40,(R0)+               ;LOAD A SPACE
004324  005303                        DEC     R3
004326  001374                        BNE     1$                      ;BR IF NOT DONE
004330  000207                        RTS     PC                      ;EXIT
```

```
                                        ;NOW ACTUAL DISPLAY THE CHARACTER SET FRAME NOT ROTATED

004332  012737  000600  007514  FILE4A: MOV     #600,TEMPA              ;LOAD A COUNTER
004340  012737  155000  026070          MOV     #CHRRTO,BUFFER          ;DISABLE ROTATE
004346  012737  001700  026100  1$:     MOV     #MAXY-77,BUFFER+10      ;LOAD STARTING Y POINT
004354  004537  007520                  JSR     R5,MESG                 ;DISPLAY IN UPPER HALF OF SCREEN
004360  000001                          1
004362  026070                          BUFFER

004364  012737  000400  026100          MOV     #400,BUFFER+10          ;LOAD STARTING Y POINT IN THE LOWER HALF
004372  004537  007520                  JSR     R5,MESG                 ;DISPLAY IN LOWER HALF OF SCREEN
004376  000001                          1
004400  026070                          BUFFER

004402  004537  007520                  JSR     R5,MESG                 ;DISPLAY SUPER AND SUBSCRIPT IN THE MIDDLE
004406  000001                          1
004410  015072                          SUPPIC

004412  005337  007514                  DEC     TEMPA                   ;FINISHED ?
004416  001353                          BNE     1$                      ;BR IF NOT
004420  005737  007622                  TST     SWITCH                  ;TEST IF "FREEZE"
004424  001342                          BNE     FILE4A                  ;BR IF YES

                                        ;NOW DISPLAY THE CHARACTER SET FRAME ROTATED

004426  012737  000600  007514  2$:     MOV     #600,TEMPA              ;LOAD DELAY COUNTER FOR THIS HALF
004434  012737  155400  026070          MOV     #CHRRT1,BUFFER          ;ENABLE CHAR. ROTATE
004442  005037  026100                  CLR     BUFFER+10               ;RESET Y ORGIN
004446  012737  000100  026076  3$:     MOV     #100,BUFFER+6           ;LOAD X ORGIN
004454  004537  007520                  JSR     R5,MESG                 ;DISPLAY FRAME
004460  000001                          1
004462  026070                          BUFFER

004464  012737  001400  026076          MOV     #1400,BUFFER+6          ;REPOSITION THE X ORGIN
004472  004537  007520                  JSR     R5,MESG                 ;DISPLAY FRAME AT NEW ORGIN
004476  000001                          1
004500  026070                          BUFFER

004502  004537  007520                  JSR     R5,MESG                 ;DISPLAY TEXT
004506  000001                          1
004510  015120                          SUPCO

004512  005337  007514                  DEC     TEMPA                   ;FINISHED DELAY ?
004516  001353                          BNE     3$                      ;BR IF NOT
004520  005737  007622                  TST     SWITCH                  ;TEST IF FREEZE
004524  001340                          BNE     2$                      ;BR IF  YES
```

```
                                ;;*****************************************************************
                                ;*TEST 16      N      SYNC SPEED AND CHARACTER TERMINATE TEST
                                ;;*****************************************************************
004526  000004                  TST16:  SCOPE
                                NO=47516                        ;ASCII VALUE FOR "NO"
                                S40=30064                       ;  "    "       "40"
                                S30=30063                       ;  "    "       "30"

004530  012777  000377  174554          MOV     #377,@VSTERM            ;LOAD TERMINATE REG.
004536  012737  047516  015320          MOV     #NO,SYNSPD              ;LOAD SYNC ASCII VALUE
004544  012737  170000  015226          MOV     #STATSA,SYNPIC          ;LOAD NO SYNC ENABLE
004552  004537  007520          1$:     JSR     R5,MESG                 ;DISPLAY THAT FRAME WITH "NO" SYNC
004556  010000                          10000
004560  015226                          SYNPIC
004562  005737  007622                  TST     SWITCH                  ;TEST IF HOLD SET
004566  001371                          BNE     1$                      ;BR IF HOLD

004570  012777  000377  174514          MOV     #377,@VSTERM            ;LOAD TERMINATE REG.
004576  012737  030064  015320          MOV     #S40,SYNSPD             ;LOAD SYNC ASCII VALUE
004604  012737  170010  015226          MOV     #STATSA!SYNC40,SYNPIC   ;LOAD SYNC ENABLE TO 40
004612  004537  007520          2$:     JSR     R5,MESG                 ;DISPLAY THAT FRAME WITH "40" SYNC
004616  000200                          200
004620  015226                          SYNPIC
004622  005737  007622                  TST     SWITCH                  ;TEST IF HOLD SET
004626  001371                          BNE     2$                      ;BR IF HOLD

004630  012777  000377  174454          MOV     #377,@VSTERM            ;LOAD TERMINATE REG.
004636  012737  030063  015320          MOV     #S30,SYNSPD             ;LOAD ASCII VALUE OF 30
004644  012737  170004  015226          MOV     #STATSA!SYNC30,SYNPIC   ;LOAD 30 CPS ENABLE
004652  004537  007520          3$:     JSR     R5,MESG                 ;DISPLAY THAT FRAME AT "30" SYNC
004656  000200                          200
004660  015226                          SYNPIC
004662  005737  007622                  TST     SWITCH                  ;TEST IF HOLD
004666  001371                          BNE     3$                      ;BR IF HOLD
```

```
                                ;;*****************************************************************
                                ;*TEST 17       0       DASH LINES AND BLINK
                                ;;*****************************************************************
004670  000004                  TST17:  SCOPE
004672  004537  007520                  JSR     R5,MESG                 ;EXIT TO DISPLAY A FRAME
004676  020000                          20000
004700  015450                          FRME5                           ;USING THE DASH AND BLINK FRAME

                                ;;*****************************************************************
                                ;*TEST 20       P       VECTOR SPRAY (LENGTH) TEST
                                ;;*****************************************************************
004702  000004                  TST20:  SCOPE
004704  012700  026070                  MOV     #BUFFER,R0              ;LOAD BUFFER POINTER
004710  012737  041776  007700          MOV     #INTX!MAXX-1,DELTX6     ;LOAD X PRESET VALUE
004716  012737  000001  007702          MOV     #1,DELTY6               ;LOAD Y PRESET VALUE
004724  004737  005030          1$:     JSR     PC,SPRAY                ;LOAD INCREASING ANGLE VECTOR
004730  062737  000002  007702          ADD     #2,DELTY6               ;UPDATE Y LENGTH
004736  023727  007702  001777          CMP     DELTY6,#MAXY            ;TEST IF END
004744  003767                          BLE     1$                      ;BR IF NOT
004746  012737  001777  007702          MOV     #MAXY,DELTY6            ;RESET MAX Y LENGTH
004754  000407                          BR      4$
004756  162737  000002  007700  3$:     SUB     #2,DELTX6               ;REDUCE X LENGHT
004764  023727  007700  040000          CMP     DELTX6,#INTX            ;TEST IF END
004772  002403                          BLT     2$                      ;BR IF DONE
004774  004737  005030          4$:     JSR     PC,SPRAY                ;LOAD DECREASING ANGLE VECTOR
005000  000766                          BR      3$
005002  012720  173400          2$:     MOV     #DSTOP,(R0)+            ;LOAD STOP
005006  012720  160000                  MOV     #DJMP,(R0)+
005012  012720  015676                  MOV     #FRME6,(R0)+            ;RESTART DISPLAY FRAME
005016  004537  007520                  JSR     R5,MESG                 ;DISPALY PICTURE
005022  000200                          200                             ;COUNT
005024  015676                          FRME6
005026  000407                          BR      TST21           ;;BR TO NEXT TEST
005030  013720  007700          SPRAY:  MOV     DELTX6,(R0)+            ;LOAD X VECTOR LENGTH
005034  013720  007702                  MOV     DELTY6,(R0)+            ;LOAD Y VECTOR LENGTH
005040  005020                          CLR     (R0)+                   ;VECTOR BACK TO ORGIN
005042  005020                          CLR     (R0)+
005044  000207                          RTS     PC                      ;EXIT
```

```
 765                                    ;;*****************************************************************
 (3)                                    ;*TEST 21        Q       HORIZONTAL PHOSPHOR TEST
 (3)                                    ;;*****************************************************************
 (2)  005046  000004                    TST21:  SCOPE
 766  005050  005037  015752                      CLR     DELTX7
 767  005054  012737  000004  007514    1$:     MOV     #4,TEMPA                ;LOAD DELAY COUNT
 768  005062  004537  007520            2$:     JSR     R5,MESG                 ;EXIT TO DISPLAY A FRAME
 769  005066  000004                              4
 770  005070  015750                              FRME10                          ;USING THE HORIZ FRAME
 771  005072  004537  007520                      JSR     R5,MESG                 ;EXIT TO DISPLAY A FRAME
 772  005076  000001                              1
 773  005100  016764                              FRM10                           ;USING THE PERIMETER BOX
 774  005102  005737  007622                      TST     SWITCH                  ;TEST THE "MOTION-SWITCH"
 775  005106  001362                              BNE     1$                      ;BR IF FREEZE THE MOVEMENT
 776  005110  005337  007514                      DEC     TEMPA                   ;DELAY DONE ?
 777  005114  100362                              BPL     2$                      ;BR IF NOT
 778  005116  005237  015752                      INC     DELTX7                  ;UPDATE THE X ORIGIN
 779  005122  022737  001777  015752            CMP     #1777,DELTX7            ;TEST IF THE END
 780  005130  001351                              BNE     1$                      ;BR IF NOT
 781                                    ;;*****************************************************************
 (3)                                    ;*TEST 22        R       VERTICAL PHOSPHOR TEST
 (3)                                    ;;*****************************************************************
 (2)  005132  000004                    TST22:  SCOPE
 782  005134  005037  016512                      CLR     DELT11
 783  005140  005037  016232                      CLR     DELTY7
 784  005144  012737  000004  007514    1$:     MOV     #4,TEMPA                ;LOAD DELAY COUNT
 785  005152  004537  007520            2$:     JSR     R5,MESG                 ;EXIT TO DISPLAY A FRAME
 786  005156  000004                              4
 787  005160  016226                              FRME11                          ;USING THE VERT FRAME
 788  005162  004537  007520                      JSR     R5,MESG                 ;EXIT TO DISPLAY A FRAME
 789  005166  000001                              1
 790  005170  016764                              FRM10                           ;USING THE PERIMETER BOX
 791  005172  004537  007520                      JSR     R5,MESG                 ;DISPLAY THE MENU BOX
 792  005176  000001                              1
 793  005200  017024                              FRM11M
 794  005202  004537  007520                      JSR     R5,MESG                 ;DISPLAY THE TEST IN THE MENU
 795  005206  000004                              4
 796  005210  016506                              FRM11S                          ;DISPLAY THE 'MENU' PHOSPHOR PIC.
 797  005212  005737  007622                      TST     SWITCH                  ;TEST THE "MOTION-SWITCH"
 798  005216  001352                              BNE     1$                      ;BR IF FREEZE THE MOVEMENT
 799  005220  005337  007514                      DEC     TEMPA                   ;DELAY DONE ?
 800  005224  100352                              BPL     2$                      ;BR IF NOT
 801  005226  022737  001277  016512            CMP     #1277,DELT11            ;TEST IF AT TOP OF MENU
 802  005234  001402                              BEQ     3$                      ;BR IF YES, DONT ADVANCE THE MENU
 803  005236  005237  016512                      INC     DELT11                  ;UPDATE THE Y MENU ORGIN
 804  005242  005237  016232            3$:     INC     DELTY7                  ;UPDATE THE Y ORIGIN
 805  005246  022737  001777  016232            CMP     #1777,DELTY7            ;TEST IF THE END
 806  005254  001333                              BNE     1$                      ;BR IF NOT
```

```
                                                ;;*************************************************************
                                                ;*TEST 23        S        SHORT VECTOR AND RELATIVE POINT
                                                ;;*************************************************************
005256  000004                                  TST23:  SCOPE
005260  012700  026070                                  MOV     #BUFFER,R0              ;SET UP R0
005264  012720  114000                                  MOV     #POINT,(R0)+            ;SET UP INITIAL
005270  012720  000240                                  MOV     #240,(R0)+              ;X POSITION
005274  012720  001000                                  MOV     #MAXY+1/2,(R0)+         ;Y POSITION
005300  012720  104000                                  MOV     #SHORTV,(R0)+           ;LOAD "SHORT VECTOR"
005304  004737  005336                                  JSR     PC,LOADVT               ;LOAD THE DISPLAY PATTERN
005310  012720  130000                                  MOV     #RELATP,(R0)+           ;LOAD "RELATIVE POINT"
005314  004737  005336                                  JSR     PC,LOADVT               ;LOAD THE DISPLAY PATTERN
005320  012720  173400                                  MOV     #DSTOP,(R0)+            ;LOAD "DISPLAY STOP"
005324  012720  160000                                  MOV     #DJMP,(R0)+             ;LOAD "DISPLAY JUMP"
005330  012720  026070                                  MOV     #BUFFER,(R0)+           ;TO THE BUFFER ADDRESS
005334  000413                                          BR      FIL14A                  ;BR TO THE FRAME

005336  012737  000024  007512          LOADVT: MOV     #24,CNTR                ;LOAD A COUNTER
005344  012720  040077                  1$:     MOV     #INTX+77,(R0)+          ;LOAD A DELTA Y
005350  012720  004177                          MOV     #4177,(R0)+             ;LOAD A DELTA X,Y
005354  005337  007512                          DEC     CNTR                    ;FINISHED?
005360  001371                                  BNE     1$                      ;BR IF NOT
005362  000207                                  RTS     PC                      ;EXIT

                                        ;DISPLAY FOUR SHORT VECTOR/RELATIVE POINT OCTAGONS IN DIFFERENT QUADRANTS
005364  012737  006000  007514          FIL14A: MOV     #6000,TEMPA             ;LOAD COUNTER
005372  012737  000200  017070          1$:     MOV     #200,FRM14A             ;LOAD FIRST OCTAGON
005400  012737  000200  017072                  MOV     #200,FRM14B
005406  004537  007520                          JSR     R5,MESG                 ;DISPLAY OCT.
005412  000001                                  1
005414  017064                                  FRME14
005416  012737  001400  017070                  MOV     #1400,FRM14A            ;LOAD SECOND OCTAGON
005424  012737  000200  017072                  MOV     #200,FRM14B
005432  004537  007520                          JSR     R5,MESG                 ;DISPLAY 2ND OCT.
005436  000001                                  1
005440  017064                                  FRME14
005442  012737  001400  017070                  MOV     #1400,FRM14A            ;LOAD THIRD OCTAGON
005450  012737  001400  017072                  MOV     #MAXY-377,FRM14B
005456  004537  007520                          JSR     R5,MESG
005462  000001                                  1
005464  017064                                  FRME14
005466  012737  000200  017070                  MOV     #200,FRM14A             ;LOAD FOURTH OCTAGON
005474  012737  001400  017072                  MOV     #MAXY-377,FRM14B
005502  004537  007520                          JSR     R5,MESG                 ;DISPLAY 4TH OCT.
005506  000001                                  1
005510  017064                                  FRME14
                                        ;NOW DISPLAY THE SHORT VECTOR/RELATIVE POINT VERTICAL LINES
005512  004537  007520                          JSR     R5,MESG                 ;DISPLAY BAR
005516  000001                                  1
005520  026070                                  BUFFER
005522  005337  007514                          DEC     TEMPA                   ;FINISHED ?
005526  001321                                  BNE     1$                      ;BR IF NOT
```

```
859                                     ;;*****************************************************************
(3)                                     ;*TEST 24       T       GRAPHPLOT INCREMENT REGISTER TEST USING GRAPHPLOT X AND  Y
(3)                                     ;;*****************************************************************
(2)  005530  000004                     TST24:  SCOPE
860  005532  012737  174100  017222             MOV     #STATSB!INCR,GRPINC     ;LOAD BASIC INCREMENT VALUE
861  005540  004537  007520            1$:      JSR     R5,MESG                 ;DISPLAY FRAME
862  005544  002000                             2000
863  005546  017176                             GRAPH
864  005550  005237  017222                     INC     GRPINC                  ;UPDATE INCR. VALUE
865  005554  022737  174110  017222             CMP     #STATSB!INCR+10,GRPINC  ;TEST IF #10
866  005562  001366                             BNE     1$                      ;BR IF NOT
867
868                                     ;;*****************************************************************
(3)                                     ;*TEST 25       U       INTENSITY LEVEL AND LIGHT PEN TEST
(3)                                     ;;*****************************************************************
(2)  005564  000004                     TST25:  SCOPE
869  005566  012777  007056  173524             MOV     #RET14,@LPVCT           ;LOAD LIGHT PEN VECTOR
870  005574  113777  001243  173520             MOVB    $VECT1+1,@LPVCT1
871  005602  042777  177400  173512             BIC     #177400,@LPVCT1         ;MASK
872  005610  012737  000010  007500             MOV     #10,DSAVE1              ;SET UP COUNT
873  005616  012737  022270  022160             MOV     #PENOF0,MSOPEN          ;RESET PEN MESSAGE #0
874  005624  012737  022350  022230             MOV     #PENOF1,MS1PEN          ;RESET PEN MESSAGE #1
875  005632  012700  026070            1$:      MOV     #BUFFER,R0              ;LOAD START ADDR.
876  005636  012737  000100  007476             MOV     #100,DSAVE
877  005644  012720  117600                     MOV     #POINT!INT7,(R0)+       ;LOAD POINT
878  005650  012720  001400                     MOV     #1400,(R0)+             ;LOAD X POINT
879  005654  012720  000300                     MOV     #300,(R0)+              ;LOAD Y POINT
880  005660  004737  007020                     JSR     PC,LOADUP               ;LOAD UP THE BUFFER
881  005664  012720  173400                     MOV     #DSTOP,(R0)+            ;LOAD DSTOP
882  005670  012720  160000                     MOV     #DJMP,(R0)+             ;LOAD DJUMP
883  005674  012720  026070                     MOV     #BUFFER,(R0)+           ;LOAD RETURN ADDRESS
884  005700  005037  007474                     CLR     HITCNT                  ;CLEAR HIT COUNT
885  005704  012737  030060  022262             MOV     #30060,FRM16B-2         ;PRESET THE HIT COUNT VALUE
886  005712  012737  030060  022260             MOV     #30060,FRM16B-4
887
888  005720  005037  007472            2$:      CLR     VIEW
889  005724  004537  007520                     JSR     R5,MESG                 ;EXIT TO DISPLAY FRAME
890  005730  000004                             4
891  005732  022040                             FRME16                          ;SUB-PICTURE
892
893  005734  005237  007472                     INC     VIEW
894  005740  004537  007520                     JSR     R5,MESG                 ;EXIT TO DISPLAY FRAME
895  005744  000001                             1
896  005746  026070                             BUFFER
897
898  005750  005337  007476                     DEC     DSAVE                   ;FINISHED ?
899  005754  001361                             BNE     2$                      ;BR IF NOT MINI-LOOP
900  005756  005337  007500                     DEC     DSAVE1                  ;FINISHED ?
901  005762  001323                             BNE     1$                      ;BR IF NOT
902  005764  013777  001322  173326             MOV     LPVCT1,@LPVCT           ;RESET VECTOR
903  005772  005077  173324                     CLR     @LPVCT1
904
```

```
                                 ;;*******************************************************************
                                 ;*TEST 26        W         DYNAMIC EXT. DISPLAY STOP
                                 ;;*******************************************************************
005776  000004                   TST26:  SCOPE
006000  013700  001242                   MOV     $VECT1,R0               ;LOAD VECTOR POINTER
006004  042700  160000                   BIC     #160000,R0              ;MASK
006010  012720  006206                   MOV     #4$,(R0)+               ;LOAD STOP VECTOR
006014  012720  000200                   MOV     #200,(R0)+
006020  012720  006272                   MOV     #BAD1,(R0)+             ;LOAD UNEXPT. INTR
006024  012720  000340                   MOV     #340,(R0)+
006030  012720  006304                   MOV     #BAD2,(R0)+             ;LOAD UNEXPT. INTR
006034  012720  000340                   MOV     #340,(R0)+
006040  012720  006316                   MOV     #BAD3,(R0)+             ;LOAD UNEXPT. INTR
006044  012720  000340                   MOV     #340,(R0)+
006050  005037  177776                   CLR     PSW
                                 ;START DISPLAY AND DELAY
006054  012777  024570  173200           MOV     #FRME17,@DPC            ;START DISPLAY
006062  012702  000400                   MOV     #400,R2                 ;LOAD TIMER COUNTER
006066  012700  001334           1$:     MOV     #BEGIN,R0               ;LOAD RANDOM NUMBER POINTER
006072  112001                   2$:     MOVB    (R0)+,R1                ;GET A RANDOM NUMBER
006074  042701  177700                   BIC     #177700,R1              ;MASK OFF OTHER BITS
006100  022700  026070                   CMP     #BUFFER,R0              ;TEST IF DONE
006104  001770                           BEQ     1$                      ;BR BACK
006106  005037  007472                   CLR     DIDINT                  ;CLEAR "DID INTERRUPT " FLAG
006112  005301                   3$:     DEC     R1                      ;DELAY
006114  001376                           BNE     3$
                                 ;NOW SET EXT. STOP FLAG
006116  052777  000200  173150           BIS     #BIT7,@DSR1             ;SET EXT. STOP FLAG
006124  012703  010000                   MOV     #BIT12,R3               ;LOAD DELAY COUNTER
006130  017737  173126  001126   7$:     MOV     @DPC,$BDDAT             ;TEST IF DPC IS OUT OF RANGE
006136  012737  024570  001124           MOV     #FRME17,$GDDAT          ;LOAD LOW LIMIT
006144  023737  001124  001126           CMP     $GDDAT,$BDDAT           ;COMAPRE
006152  101066                           BHI     BAD4                    ;BR IF TOO LOW
006154  012737  025244  001124           MOV     #FRM17F+4,$GDDAT        ;LOAD HIGH LIMIT
006162  023737  001124  001126           CMP     $GDDAT,$BDDAT           ;COMAPRE
006170  103463                           BLO     BAD5                    ;BR IF TOO HIGH
006172  005737  007472                   TST     DIDINT                  ;TEST IF EXT. STOP INTR. OCCURRED
006176  001335                           BNE     2$                      ;BR IF YES
006200  005303                           DEC     R3                      ;DELAY
006202  001352                           BNE     7$                      ;BR AND TEST DPC VALUE
006204  000426                           BR      BAD0                    ;NO EXT. STOP INTR. REPORT ERROR
006206  105777  173062           4$:     TSTB    @DSR1                   ;TEST IF EXT. STOP FLAG
006212  100407                           BMI     5$                      ;BR IF EXT. STOP
006214  005777  173044                   TST     @DSR                    ;TEST FOR DISPLAY STOP
006220  100053                           BPL     BAD6                    ;BR IF NOT
006222  012777  024570  173032           MOV     #FRME17,@DPC            ;START DPU IF NOT EXT. STOP
006230  000002                           RTI                             ;RETURN
006232  005302                   5$:     DEC     R2                      ;FINISHED ?
006234  001407                           BEQ     6$                      ;BR IF DONE
006236  012777  000001  173016           MOV     #BIT0,@DPC              ;RESUME THE DPU IF EXT. STOP AND NOT FILISHED LO
006244  052737  000001  007472           BIS     #1,DIDINT               ;SET EXT. STOP FLAG DID INTR.
006252  000002                           RTI                             ;RETURN
006254  022626                   6$:     CMP     (SP)+,(SP)+             ;CLEAN THE STACK
006256  000137  006434                   JMP     $EOP
```

```
                                        ;BR HERE IF AN ERROR OCCURRED
                                        
006262  012737  025276  025264  BAD0:   MOV     #WHY0,WHY               ;INDICATE NO EXT. STOP INTERRUPT
006270  000432                         BR      BADDON
006272  012737  025332  025264  BAD1:   MOV     #WHY1,WHY               ;INDICATE UNEXPECTED INTR.
006300  022626                         CMP     (SP)+,(SP)+
006302  000425                         BR      BADDON
006304  012737  025376  025264  BAD2:   MOV     #WHY2,WHY               ;INDICATE UNEXPECTED INTR.
006312  022626                         CMP     (SP)+,(SP)+
006314  000420                         BR      BADDON
006316  012737  025442  025264  BAD3:   MOV     #WHY3,WHY               ;INDICATE UNEXPECTED INTR.
006324  022626                         CMP     (SP)+,(SP)+
006326  000413                         BR      BADDON
006330  012737  025506  025264  BAD4:   MOV     #WHY4,WHY               ;INDICATE DPC WAS TOO LOW
006336  000407                         BR      BADDON
006340  012737  025526  025264  BAD5:   MOV     #WHY5,WHY               ;INDICATE DPC WAS TOO HIGH
006346  000403                         BR      BADDON
006350  012737  025550  025264  BAD6:   MOV     #WHY6,WHY               ;INDICATE DONE INTR. BUT NO FLAG
                                        
                                        
006356  017737  172700  006426  BADDON: MOV     @DPC,PCERR              ;SAVE DPC
006364  017737  172674  006430          MOV     @DSR,SRERR              ;SAVE SR
006372  017737  172676  006432          MOV     @DSR1,SR1ERR            ;SAVE SR1
006400  000240                         NOP
006402  000240                         NOP
006404  012777  007546  172702          MOV     #MESGAA,@DDONE          ;LOAD DISPLAY STOP VECTOR
006412  004537  007520                 JSR     R5,MESG                 ;DISPLAY ERROR MESSAGE
006416  040000                         BIT14
006420  025242                         FRM17E
006422  000005                         RESET
006424  000403                         BR      $EOP                    ;END OF PASS
                                        
006426  000000                 PCERR:  0                               ;D.P.C. UPON ERROR
006430  000000                 SRERR:  0                               ;SR UPON ERROR
006432  000000                 SR1ERR: 0                               ;SR1 UPON ERROR
```

```
                                        .SBTTL

                                        .SBTTL  END OF PASS ROUTINE

                                        ;;*****************************************************************
                                        ;*INCREMENT THE PASS NUMBER ($PASS)
                                        ;*IF THERES A MONITOR GO TO IT
                                        ;*IF THERE ISN'T JUMP TO RESTAT

006434                                  $EOP:
006434  000004                                 SCOPE
006436  005037  001102                         CLR     $TSTNM          ;;ZERO THE TEST NUMBER
006442  005237  001200                         INC     $PASS           ;;INCREMENT THE PASS NUMBER
006446  042737  100000  001200                 BIC     #100000,$PASS   ;;DON'T ALLOW A NEG. NUMBER
006454  005327                                 DEC     (PC)+           ;;LOOP?
006456  000001                          $EOPCT: .WORD   1
006460  003013                                 BGT     $DOAGN          ;;YES
006462  012737                                 MOV     (PC)+,@(PC)+    ;;RESTORE COUNTER
006464  000001                          $ENDCT: .WORD   1
006466  006456                                         $EOPCT
006470  013700  000042                  $GET42: MOV     @#42,R0         ;;GET MONITOR ADDRESS
006474  001405                                 BEQ     $DOAGN          ;;BRANCH IF NO MONITOR
006476  000005                                 RESET                   ;;CLEAR THE WORLD
006500  004710                          $ENDAD: JSR     PC,(R0)         ;;GO TO MONITOR
006502  000240                                 NOP                     ;;SAVE ROOM
006504  000240                                 NOP                     ;;FOR
006506  000240                                 NOP                     ;;ACT11
006510                                  $DOAGN:
006510  000137                                 JMP     @(PC)+          ;;RETURN
006512  001516                          $RTNAD: .WORD   RESTAT
```

```
 1005                                   ;;*****************************************************************
  (3)                                   ;*TEST 27       V       KEYBOARD CHARACTER ECHO LOOP
  (3)                                   ;;*****************************************************************
  (2)  006514  000004                   TST27:  SCOPE
 1006  006516  012737  030060  024532           MOV     #30060,ECODEV-4         ;PRESET READOUT TO 00
 1007  006524  012737  030060  024534           MOV     #30060,ECODEV-2         ;PRESET READOUT TO 00
 1008  006532  012700  026070           20$:    MOV     #BUFFER,R0              ;LOAD BUFFER POINTER
 1009  006536  012701  001000                   MOV     #512.,R1                ;LOAD CHARACTER COUNT
 1010  006542  005020                   1$:     CLR     (R0)+                   ;CLEAR THE BUFFER
 1011  006544  005301                           DEC     R1                      ;FINISHED ?
 1012  006546  001375                           BNE     1$                      ;BR IF NOT
 1013  006550  012720  160000                   MOV     #DJMP,(R0)+             ;LOAD JUMP RETURN TO START OF BUFFER
 1014  006554  012720  024400                   MOV     #ECHOFR,(R0)+           ;  THE ECHO FRAME
 1015  006560  012737  161010  024540           MOV     #161010,ECHJMP          ;PRESET JUMP
 1016  006566  005037  007472                   CLR     SHIFTO                  ;CLEAR SHIFT IND.
 1017  006572  012700  026070                   MOV     #BUFFER,R0              ;LOAD BUFFER POINTER
 1018  006576  012701  002000                   MOV     #1024.,R1               ;LOAD CHARACTER COUNT
 1019  006602  012777  006636  172446           MOV     #10$,@TKBVCT            ;LOAD INTR. RETURN
 1020  006610  012777  000200  172442           MOV     #200,@TKBVTI            ; LOAD RETURN INTR. LEVEL
 1021  006616  052777  000100  172320           BIS     #BIT6,@$TKS             ;ENABLE KEYBOARD INTR.
 1022  006624  004537  007520           3$:     JSR     R5,MESG                 ;DISPLAY FRAME AND BUFFER
 1023  006630  000001                           1
 1024  006632  024400                           ECHOFR                          ;ADDRESS OF SUB-PICTURE
 1025  006634  000736                           BR      20$                     ;BR UPON EXT. STOP INTERRUPT
 1026                                  ;RETURN HERE UPON KEYBOARD INTR. <D.P.U. SHOULD STILL BE RUNNING>
 1027  006636  017703  172304           10$:    MOV     @$TKB,R3                ;READ KEYBOARD DATA
 1028  006642  042703  177600                   BIC     #177600,R3              ;MASK TO LOWER 7 BITS
 1029  006646  005301                           DEC     R1                      ;FINISHED INPUTING MAX. CHARS ?
 1030  006650  001443                           BEQ     12$                     ;BR IF DONE MAX CHARACTERS INPUT
 1031  006652  022703  000003                   CMP     #3,R3                   ;TEST IF CHAR WAS A CTRL C ?
 1032  006656  001002                           BNE     11$                     ;BR IF NOT
 1033  006660  000137  006434                   JMP     $EOP                    ;REPORT END OF PASS AND START OVER
 1034  006664  022703  000016           11$:    CMP     #16,R3                  ;TEST FOR SHIFT OUT CODE
 1035  006670  001005                           BNE     4$                      ;BR IF NOT
 1036  006672  005237  007472                   INC     SHIFTO                  ;SET SHIFT OUT FLAG
 1037  006676  012737  164000  024540           MOV     #DNOP,ECHJMP            ;NOP THE BYPASS DISP. JMP
 1038  006704  005737  007472           4$:     TST     SHIFTO          ;TEST IF SHIFT OUT
 1039  006710  001415                           BEQ     2$                      ;BR IF NOT
 1040  006712  022703  000017                   CMP     #17,R3                  ;TEST FOR SHIFT IN CODE
 1041  006716  001005                           BNE     5$                      ;BR IF NOT
 1042  006720  005037  007472                   CLR     SHIFTO                  ;CLEAR SHIFT OUT FLAG
 1043  006724  012737  161010  024540           MOV     #161010,ECHJMP          ;LOAD BYPASS DISP. JMP
 1044  006732  122703  000037           5$:     CMPB    #37,R3                  ;TEST IF TOO BIG
 1045  006736  100002                           BPL     2$                      ;BR IF NOT
 1046  006740  042703  177740                   BIC     #177740,R3              ;MASK OFF BITS
 1047  006744  110320                   2$:     MOVB    R3,(R0)+                ;LOAD CHARACTER INTO NEXT BUFFER LOC.
 1048  006746  012702  024536                   MOV     #ECODEV,R2              ;LOAD POINTER TO ASCII CHARACTER VALUE WAS = "
 1049  006752  004737  007624                   JSR     PC,KBCHR                ;CONVERT CHARACTER VALUE TO OCTAL
 1050  006756  000002                           RTI                             ;RETURN TO WAIT
 1051  006760  022626                   12$:    CMP     (SP)+,(SP)+             ;ADJUST STACK
 1052  006762  013702  001262                   MOV     DPC,R2                  ;GET DPC ADDRESS
 1053  006766  052762  000200  000012           BIS     #BIT7,12(R2)            ;EXTERNAL STOP TO DISPLAY
 1054  006774  000656                           BR      20$                     ;CLEAR THE BUFFER AND START AGAIN
```

```
                                                .SBTTL  SUBROUTINE FOR VERT. LIGHT PEN FIELD OF VIEW
006776  012701  000030          LOADAC: MOV     #24.,R1                 ;LOAD COUNT
007002  012720  130000                  MOV     #RELATP,(R0)+           ;LOAD RELATIVE POINT
007006  012720  040004          1$:     MOV     #INTX+4,(R0)+           ;LOAD INTEN BIT
007012  005301                          DEC     R1                      ;FINISHED ?
007014  001374                          BNE     1$                      ;BR IF NOT
007016  000207                          RTS     PC                      ;EXIT

                                                .SBTTL  SUBROUTINE FOR HORIZ. LIGHT PEN FIELD OF VIEW
007020  012737  000030  007512  LOADUP: MOV     #24.,CNTR               ;LOAD COUNT
007026  004737  006776          1$:     JSR     PC,LOADAC               ;LOAD ACCROSS
007032  012720  110000                  MOV     #LONGV,(R0)+            ;LOAD LONG VECTOR
007036  012720  000004                  MOV     #4,(R0)+                ;LOAD VECTOR OVER
007042  012720  020140                  MOV     #MINUSX+140,(R0)+       ;AND UP
007046  005337  007512                  DEC     CNTR
007052  001365                          BNE     1$                      ;BR IF NOT DONE
007054  000207                          RTS     PC                      ;EXIT

                                                .SBTTL  LIGHT-PEN INTERRUPT SERVICE
007056  017737  172222  001126  RET14:  MOV     @VSCONS,$BDDAT          ;READ CONSOLE STATUS REG
007064  017737  172176  007502          MOV     @XPOS,DSAVE2            ;READ X POSITION
007072  017737  172172  007504          MOV     @YPOS,DSAVE3            ;READ Y POSITION
007100  042737  176000  007502          BIC     #176000,DSAVE2          ;MASK HIGH SIX BITS
007106  042737  176000  007504          BIC     #176000,DSAVE3          ;
007114  005037  007470                  CLR     40$                     ;CLEAR SWITCH FLAG HAPPEN LOC.
007120  032737  000100  001126          BIT     #BIT6,$BDDAT            ;TEST IF CONSOLE #1 SWITCH FLAG
007126  001405                          BEQ     3$                      ;BR IF NOT
007130  012737  022350  022230          MOV     #PENOF1,MS1PEN          ;INFORM PEN OF ON #1 SET
007136  005237  007470                  INC     40$                     ;SET SW HAPPENED FLAG
007142  032737  000200  001126  3$:     BIT     #BIT7,$BDDAT            ;TEST IF CONSOLE #1 SWITCH FLAG
007150  001405                          BEQ     4$                      ;BR IF NOT
007152  012737  022400  022230          MOV     #PENON1,MS1PEN          ;INFORM PEN ON #1 SET
007160  005237  007470                  INC     40$                     ;SET SW HAPPENED FLAG
007164  032737  010000  001126  4$:     BIT     #BIT12,$BDDAT           ;TEST IF CONSOLE #0 SWITCH FLAG
007172  001405                          BEQ     5$                      ;BR IF NOT
007174  012737  022270  022160          MOV     #PENOF0,MSOPEN          ;INFORM PEN OFF #0 SET
007202  005237  007470                  INC     40$                     ;SET SW HAPPENED FLAG
007206  032737  020000  001126  5$:     BIT     #BIT13,$BDDAT           ;TEST IF CONSOLE #0 SET
007214  001405                          BEQ     6$                      ;BR IF NOT
007216  012737  022320  022160          MOV     #PENON0,MSOPEN          ;INFORM PEN ON #0 SET
007224  005237  007470                  INC     40$                     ;SET SW HAPPENED FLAG
007230  005737  007470          6$:     TST     40$                     ;TEST IF SWITCH FUNCTION
007234  001003                          BNE     12$                     ;BR IF YES
007236  005737  007472                  TST     VIEW                    ;TEST IF FIELD OF VIEW
007242  001046                          BNE     20$                     ;BR IF YES
007244  032737  040000  001126  12$:    BIT     #BIT14,$BDDAT           ;TEST IF PEN FLAG #0 SET
007252  001414                          BEQ     7$                      ;BR IF NOT
007254  013703  007502                  MOV     DSAVE2,R3               ;LOAD R3
007260  012702  022134                  MOV     #DLT14A+4,R2            ;LOAD ADDRESS
007264  004737  007624                  JSR     PC,KBCHR                ;LOAD X READOUT
007270  013703  007504                  MOV     DSAVE3,R3               ;LOAD R3
007274  012702  022146                  MOV     #DLT14B+4,R2            ;LOAD ADDRESS
007300  004737  007624                  JSR     PC,KBCHR                ;LOAD Y READOUT
007304  032737  000400  001126  7$:     BIT     #BIT8,$BDDAT            ;TEST IF PEN #1 FLAG
```

```
                                        BEQ     10$                     ;BR IF NOT
                                        MOV     DSAVE2,R3               ;GET X VALUE
                                        MOV     #DLT14C+4,R2            ;LOAD POINTER
                                        JSR     PC,KBCHR                ;CONVERT TO ASCII
                                        MOV     DSAVE3,R3               ;GET Y VALUE
                                        MOV     #DLT14D+4,R2            ;LOAD POINTER
                                        JSR     PC,KBCHR                ;CONVERT TO ASCII
                                10$:    CMP     (SP)+,(SP)+
                                        MOV     #1,@DPC         ;SINGLE STEP THE DISPLAY
                                        JMP     MESGA

                                ;COME HERE IF LIGHT-PEN HIT DURING THE FIELD OF VIEW SUB-PICTURE

                                20$:    INC     HITCNT                  ;UPDATE COUNT
                                        MOV     HITCNT,R3               ;LOAD COUNT #
                                        MOV     #FRM16B,R2              ;LOAD MESSAGE POINTER
                                        JSR     PC,KBCHR                ;CONVERT TO ASCII
                                        CLR     R1
                                        CLR     R2
                                        MOV     DSAVE2,R0               ;GET X AXIS
                                        SUB     #1400,R0                ;GET A BASE ADDRESS
                                        ASR     R0
                                        ASR     R0
                                        BEQ     30$
                                21$:    ADD     #70,R1                  ;UPDATE OFFSET
                                        DEC     R0
                                        BNE     21$                     ;BR UNTIL DONE
                                30$:    MOV     DSAVE3,R0               ;GET Y AXIS
                                        SUB     #304,R0                 ;MAKE BASE ADDRESS
                                        ASR     R0
                                        ASR     R0                      ;SHIFT RIGHT
                                        BEQ     32$
                                31$:    ADD     #2,R1
                                        DEC     R0
                                        BNE     31$
                                32$:    BIC     #INTX,BUFFER+10(R1)     ;CLEAR THE BIT
                                        BR      10$

                                40$:    0

                                SHIFTO:
                                DIDINT:
                                VIEW:   0
                                HITCNT: 0
                                DSAVE:  0
                                DSAVE1: 0
                                DSAVE2: 0
                                DSAVE3: 0
                                HOLD:   0
                                TSAVE:  0
                                CNTR:   0
                                TEMPA:  0
                                TEMPB:  0
```

```
                                                .SBTTL  DISPLAY SUB-ROUTINE
                                        ; ARG ARE LOOP COUNT AND DISPLAY BUFFER ADDRESS
                                        ; UPON INTERRUPT , DEC LOOP COUNT
                                        ;  RESUME DISPLAY IF NOT 0
                                        ;  RTS R5 IF COMPLETED

007520  012537  007616          MESG:   MOV     (R5)+,COUNT
007524  012537  007620                  MOV     (R5)+,FILE
007530  013777  007620  171524          MOV     FILE,@DPC       ;START DISPLAY
007536  005037  177776          MESGA:  CLR     PSW
007542  000001                          WAIT
007544  000774                          BR      MESGA                   ;LOOP BACK
                                        ;
                                        ;RETURN HERE UPON STOP INTERRUPT
                                        ;
007546  022626                  MESGAA: CMP     (SP)+,(SP)+             ;ADJUST STACK
007550  005337  007616                  DEC     COUNT                   ;FINISHED LOOPING ?
007554  001404                          BEQ     1$                      ;BR IF YES
007556  012777  000001  171476          MOV     #1,@DPC         ;SINGLE STEP THE DISPLAY
007564  000764                          BR      MESGA
007566  105777  171346          1$:     TSTB    @SWR                    ;TEST IF KEYBOARD CONTROL ?
007572  100410                          BMI     MESGBA                  ;BR IF YES
007574  005037  007622                  CLR     SWITCH
007600  032777  001000  171332          BIT     #BIT9,@SWR              ;TEST SWITCH BIT 9
007606  001402                          BEQ     MESGBA
007610  005137  007622                  COM     SWITCH                  ;SET FLAG IF SWR 9 = 1
007614  000205                  MESGBA: RTS     R5                      ;EXIT
007616  000000                  COUNT:  0
007620  000000                  FILE:   0
007622  000000                  SWITCH: 0

                                                .SBTTL  UPDATE OCTAL READOUT OF THE X-Y FOR LIGHT PEN HIT

007624  004737  007664          KBCHR:  JSR     PC,10$                  ;LOAD BITS
007630  110442                          MOVB    R4,-(R2)                ;SAVE BITS
007632  004737  007656                  JSR     PC,11$                  ;MOVE BITS
007636  110442                          MOVB    R4,-(R2)                ;SAVE BITS
007640  004737  007656                  JSR     PC,11$                  ;MOVE BITS
007644  110442                          MOVB    R4,-(R2)                ;SAVE BITS
007646  004737  007656                  JSR     PC,11$
007652  110442                          MOVB    R4,-(R2)
007654  000207                          RTS     PC
007656  006003          11$:    ROR     R3
007660  006003                  ROR     R3
007662  006003                  ROR     R3
007664  010304          10$:    MOV     R3,R4                   ;LOAD R4
007666  042704  177770                  BIC     #177770,R4              ;MASK BITS
007672  062704  000060                  ADD     #60,R4                  ;MAKE A NUMBER
007676  000207                          RTS     PC

007700  000000          DELTX6: 0
007702  000000          DELTY6: 0
```

```
                                                .SBTTL  X - Y POSITIONS FOR THE SHORT TERM DRIFT TEST
                                                
007704  000000  000000  000000  TABA:   .WORD   0,0,0,0,0,0
007712  000000  000000  000000
007720  001000  001000                  .WORD   1000,1000
007724  000000  001777  000000          .WORD   0,MAXY,0,MAXY,0,MAXY
007732  001777  000000  001777
007740  001000  001000                  .WORD   1000,1000
007744  001777  001777  001777          .WORD   MAXX,MAXY,MAXX,MAXY,MAXX,MAXY
007752  001777  001777  001777
007760  001000  001000                  .WORD   1000,1000
007764  001777  000000  001777          .WORD   MAXX,0,MAXX,0,MAXX,0
007772  000000  001777  000000
010000  001000  001000                  .WORD   1000,1000
010004  100000                          BIT15
```

```
                                FRMED:  POINT
                                        0
                                        MAXY-277
                                        CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
                                        CHAR
1268  010020  051526  033055  020060    .ASCII  /VS-60 VISUAL DISPLAY TEST  <MD-11-DZVSDB> /
      010026  044526  052523  046101
      010034  042040  051511  046120
      010042  054501  052040  051505
      010050  020124  036040  042115
      010056  030455  026461  055104
      010064  051526  041104  020076
1269  010072     015     012     012    .BYTE   15,12,12
1270  010075     040  020040  044504    .ASCII  /   DIRECTORY OF THE TESTS/
      010102  042522  052103  051117
      010110  020131  043117  052040
      010116  042510  052040  051505
      010124  051524
1271  010126     015     012     012    .BYTE   15,12,12
1272  010131     101  036440  030040    .ASCII  /A = 01 = DIRECTORY FRAME/
      010136  020061  020075  044504
      010144  042522  052103  051117
      010152  020131  051106  046501
      010160     105
1273  010161     015     012            .BYTE   15,12
1274  010163     102  036440  030040    .ASCII  /B = 02 = ASTIGMATISM AND SETTLING/
      010170  020062  020075  051501
      010176  044524  046507  052101
      010204  051511  020115  047101
      010212  020104  042523  052124
      010220  044514  043516
1275  010224     015     012            .BYTE   15,12
1276  010226  020103  020075  031460    .ASCII  /C = 03 = SHORT TERM DRIFT/
      010234  036440  051440  047510
      010242  052122  052040  051105
      010250  020115  051104  043111
      010256     124
1277  010257     015     012            .BYTE   15,12
1278  010261     104  036440  030040    .ASCII  /D = 04 = MINOR AXIS GAIN/
      010266  020064  020075  044515
      010274  047516  020122  054101
      010302  051511  043440  044501
      010310     116
1279  010311     015     012            .BYTE   15,12
1280  010313     105  036440  030040    .ASCII  /E = 05 = MAJOR AXIS OFFSET/
      010320  020065  020075  040515
      010326  047512  020122  054101
      010334  051511  047440  043106
      010342  042523     124
1281  010345     015     012            .BYTE   15,12
1282  010347     106  036440  030040    .ASCII  /F = 06 = VECTOR LENGTH GAIN/
      010354  020066  020075  042526
      010362  052103  051117  046040
      010370  047105  052107  020110
```

```
        010376  040507  047111
1283    010402     015     012                  .BYTE   15,12
1284    010404  020107  020075  033460          .ASCII  /G = 07 = PINCUSHION/
        010412  036440  050040  047111
        010420  052503  044123  047511
        010426     116
1285    010427     015     012                  .BYTE   15,12
1286    010431     110  036440  030440          .ASCII  /H = 10 = OCTAGONS AND CIRCLES/
        010436  020060  020075  041517
        010444  040524  047507  051516
        010452  040440  042116  041440
        010460  051111  046103  051505
1287    010466     015     012                  .BYTE   15,12
1288    010470  020111  020075  030461          .ASCII  /I = 11 = SCISSORING AND VECTOR SCALES/
        010476  036440  051440  044503
        010504  051523  051117  047111
        010512  020107  047101  020104
        010520  042526  052103  051117
        010526  051440  040503  042514
        010534     123
1289    010535     015     012                  .BYTE   15,12
1290    010537     112  036440  030440          .ASCII  /J = 12 = X AND Y DYNAMIC OFFSET TEST/
        010544  020062  020075  020130
        010552  047101  020104  020131
        010560  054504  040516  044515
        010566  020103  043117  051506
        010574  052105  052040  051505
        010602     124
1291    010603     015     012                  .BYTE   15,12
1292    010605     113  036440  030440          .ASCII  /K = 13 = CHARACTER SCALE/
        010612  020063  020075  044103
        010620  051101  041501  042524
        010626  020122  041523  046101
        010634     105
1293    010635     015     012                  .BYTE   15,12
1294    010637     114  036440  030440          .ASCII  /L = 14 = CHARACER QUALITY AND CHARACTER ROTATE/
        010644  020064  020075  044103
        010652  051101  041501  051105
        010660  050440  040525  044514
        010666  054524  040440  042116
        010674  041440  040510  040522
        010702  052103  051105  051040
        010710  052117  052101     105
1295    010715     015     012                  .BYTE   15,12
1296    010717     115  036440  030440          .ASCII  /M = 15 = CHARACTER SET, SUPER AND SUBSCRIPT, AND ITALIC CHARACTERS/
        010724  020065  020075  044103
        010732  051101  041501  042524
        010740  020122  042523  026124
        010746  051440  050125  051105
        010754  040440  042116  051440
        010762  041125  041523  044522
        010770  052120  020054  047101
        010776  020104  052111  046101
        011004  041511  041440  040510
```

```
        011012  040522  052103  051105
        011020     123
 1297   011021     015     012                .BYTE   15,12
 1298   011023     116  036440  030440        .ASCII  /N = 16 = SYNC SPEED AND CHARACTER TERMINATE/
        011030  020066  020075  054523
        011036  041516  051440  042520
        011044  042105  040440  042116
        011052  041440  040510  040522
        011060  052103  051105  052040
        011066  051105  044515  040516
        011074  042524
 1299   011076     015     012                .BYTE   15,12
 1300   011100  020117  020075  033461        .ASCII  /O = 17 = DASH LINES AND BLINK/
        011106  036440  042040  051501
        011114  020110  044514  042516
        011122  020123  047101  020104
        011130  046102  047111     113
 1301   011135     015     012                .BYTE   15,12
 1302   011137     120  036440  031040        .ASCII  /P = 20 = VECTOR LENGTH/
        011144  020060  020075  042526
        011152  052103  051117  046040
        011160  047105  052107     110
 1303   011165     015     012                .BYTE   15,12
 1304   011167     121  036440  031040        .ASCII  /Q = 21 = HORIZONTAL PHOSPHOR TEST/
        011174  020061  020075  047510
        011202  044522  047532  052116
        011210  046101  050040  047510
        011216  050123  047510  020122
        011224  042524  052123
 1305   011230     015     012                .BYTE   15,12
 1306   011232  020122  020075  031062        .ASCII  /R = 22 = VERTICAL PHOSPHOR TEST/
        011240  036440  053040  051105
        011246  044524  040503  020114
        011254  044120  051517  044120
        011262  051117  052040  051505
        011270     124
 1307   011271     015     012                .BYTE   15,12
 1308   011273     123  036440  031040        .ASCII  /S = 23 = SHORT VECTORS AND RELATIVE POINT/
        011300  020063  020075  044123
        011306  051117  020124  042526
        011314  052103  051117  020123
        011322  047101  020104  042522
        011330  040514  044524  042526
        011336  050040  044517  052116
 1309   011344     015     012                .BYTE   15,12
 1310   011346  020124  020075  032062        .ASCII  /T = 24 = GRAPHPLOT X AND GRAPHPLOT Y TEST/
        011354  036440  043440  040522
        011362  044120  046120  052117
        011370  054040  040440  042116
        011376  043440  040522  044120
        011404  046120  052117  054440
        011412  052040  051505     124
 1311   011417     015     012                .BYTE   15,12
 1312   011421     125  036440  031040        .ASCII  /U = 25 = INTENSITY LEVEL AND LIGHT PEN TESTS/
```

```
        011426  020065  020075  047111
        011434  042524  051516  052111
        011442  020131  042514  042526
        011450  020114  047101  020104
        011456  044514  044107  020124
        011464  042520  020116  042524
        011472  052123     123
1313    011475     015     012                  .BYTE   15,12
1314    011477     126  036440  031040          .ASCII  /V = 26 = KEYBOARD CHARACTER ECHO LOOP /
        011504  020066  020075  042513
        011512  041131  040517  042122
        011520  041440  040510  040522
        011526  052103  051105  042440
        011534  044103  020117  047514
        011542  050117     040
1315    011545     015     012                  .BYTE   15,12
1316    011547     127  036440  031040          .ASCII  /W = 27 = DYNAMIC EXT. STOP TEST /
        011554  020067  020075  054504
        011562  040516  044515  020103
        011570  054105  027124  051440
        011576  047524  020120  042524
        011604  052123     040
1317    011607     015     012     012          .BYTE   15,12,12
1318    011612  020040  052522  047502          .ASCII  /  RUBOUT (DELETE) TO REMAIN ON THE PATTERN /
        011620  052125  024040  042504
        011626  042514  042524  020051
        011634  047524  051040  046505
        011642  044501  020116  047117
        011650  052040  042510  050040
        011656  052101  042524  047122
        011664     040
1319    011665     015     012                  .BYTE   15,12
1320    011667     040  041440  020122          .ASCIZ  /  CR TO SELECT SUB-PICTURE OR STOP MOTION /
        011674  047524  051440  046105
        011702  041505  020124  052523
        011710  026502  044520  052103
        011716  051125  020105  051117
        011724  051440  047524  020120
        011732  047515  044524  047117
        011740  000040
1321    011742  164600                       CONSL1!BIT7                     ;DISABLE CONSOLE #1
1322    011744  114000                       POINT
1323    011746  000000                       0
1324    011750  000100                       100
1325    011752  100000                       CHAR
1326    011754  044124  051511  044440       .ASCII  /THIS IS CONSOLE 0 /
        011762  020123  047503  051516
        011770  046117  020105  020060
1327    011776  164200                       CONSL0!BIT7                     ;DISABLE CONSOLE #0
1328    012000  164700                       CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
1329    012002  114000                       POINT
1330    012004  000000                       0
1331    012006  000140                       140
1332    012010  100000                       CHAR
```

```
 1333  012012  044124  051511  044440          .ASCII  /THIS IS CONSOLE 1 /
       012020  020123  047503  051516
       012026  046117  020105  020061
 1334  012034  164300                          CONSL0!BIT7!BIT6
 1335  012036  173400                          DSTOP
 1336  012040  160000                          DJMP
 1337  012042  010006                          FRME0
 1338
 1339                                          .SBTTL  X AND Y POSITIONS FOR THE SETTLING TEST
 1340
 1341  012044  164700                TABB:     CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
 1342  012046  114000                          POINT
 1343  012050  041000                          INTX!BIT9
 1344  012052  001000                          BIT9
 1345  012054  040400                          INTX!BIT8
 1346  012056  000400                          BIT8
 1347  012060  040200                          INTX!BIT7
 1348  012062  000200                          BIT7
 1349  012064  040100                          INTX!BIT6
 1350  012066  000100                          BIT6
 1351  012070  040040                          INTX!BIT5
 1352  012072  000040                          BIT5
 1353  012074  040020                          INTX!BIT4
 1354  012076  000020                          BIT4
 1355  012100  040010                          INTX!BIT3
 1356  012102  000010                          BIT3
 1357  012104  040004                          INTX!BIT2
 1358  012106  000004                          BIT2
 1359  012110  040002                          INTX!BIT1
 1360  012112  000002                          BIT1
 1361  012114  040001                          INTX!BIT0
 1362  012116  000001                          BIT0
 1363  012120  040000                          INTX
 1364  012122  000000                          0
 1365
 1366  012124  041400                          INTX!1400
 1367  012126  000400                          BIT8
 1368  012130  041600                          INTX!1600
 1369  012132  000200                          BIT7
 1370  012134  041700                          INTX!1700
 1371  012136  000100                          BIT6
 1372  012140  041740                          INTX!1740
 1373  012142  000040                          BIT5
 1374  012144  041760                          INTX!1760
 1375  012146  000020                          BIT4
 1376  012150  041770                          INTX!1770
 1377  012152  000010                          BIT3
 1378  012154  041774                          INTX!1774
 1379  012156  000004                          BIT2
 1380  012160  041776                          INTX!1776
 1381  012162  000002                          BIT1
 1382  012164  041777                          INTX!1777
 1383  012166  000001                          BIT0
 1384
```

```
 1385  012170  041400                    INTX!1400
 1386  012172  001400                    1400
 1387  012174  041600                    INTX!1600
 1388  012176  001600                    1600
 1389  012200  041700                    INTX!1700
 1390  012202  001700                    1700
 1391  012204  041740                    INTX!1740
 1392  012206  001740                    1740
 1393  012210  041760                    INTX!1760
 1394  012212  001760                    1760
 1395  012214  041770                    INTX!1770
 1396  012216  001770                    1770
 1397  012220  041774                    INTX!1774
 1398  012222  001774                    1774
 1399  012224  041776                    INTX!1776
 1400  012226  001776                    1776
 1401  012230  041777                    INTX!1777
 1402  012232  001777                    1777
 1403
 1404  012234  040400                    INTX!BIT8
 1405  012236  001400                    1400
 1406  012240  040200                    INTX!BIT7
 1407  012242  001600                    1600
 1408  012244  040100                    INTX!BIT6
 1409  012246  001700                    1700
 1410  012250  040040                    INTX!BIT5
 1411  012252  001740                    1740
 1412  012254  040020                    INTX!BIT4
 1413  012256  001760                    1760
 1414  012260  040010                    INTX!BIT3
 1415  012262  001770                    1770
 1416  012264  040004                    INTX!BIT2
 1417  012266  001774                    1774
 1418  012270  040002                    INTX!BIT1
 1419  012272  001776                    1776
 1420  012274  040001                    INTX!BIT0
 1421  012276  001777                    1777
 1422  012300  173400                    DSTOP
 1423  012302  160000                    DJMP
 1424  012304  012044                    TABB                    ;RETURN ADDRESS
```

```
                                ;FILE 2 (ANALOG TUNE-UP TEST )

012306  164700                  FRME2:  CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
                                ;INTENSIFY A POINT 3 TIMES IN EACH CORNER
012310  114000                          POINT                           ;LOWER LEFT
012312  040000                          INTX
012314  000000                          0
012316  040000                          INTX
012320  000000                          0
012322  040000                          INTX
012324  000000                          0
012326  041777                          INTX!MAXX                       ;LOWER RIGHT
012330  000000                          0
012332  041777                          INTX!MAXX
012334  000000                          0
012336  041777                          INTX!MAXX
012340  000000                          0
012342  041777                          INTX!MAXX                       ;UPPER RIGHT
012344  001777                          MAXY
012346  041777                          INTX!MAXX
012350  001777                          MAXY
012352  041777                          INTX!MAXX
012354  001777                          MAXY
012356  040000                          INTX                            ;UPPER LEFT
012360  001777                          MAXY
012362  040000                          INTX
012364  001777                          MAXY
012366  040000                          INTX
012370  001777                          MAXY
                                ;NOW DRAW THE OUTER REF. BOX
012372  114000                  PICST0: POINT
012374  000000                          0
012376  000000                          0
012400  110000                          LONGV
012402  041777                          INTX!MAXX                       ; +X, +Y
012404  000000                          0
012406  040000                          INTX                            ; +X, +Y
012410  001777                          MAXY
012412  061777                          INTX!MINUSX!MAXX                ; -X, +Y
012414  000000                          0
012416  040000                          INTX                            ; +X, -Y
012420  021777                          MINUSY!MAXY
                                ;NOW RE-DO THE BOX WITH NEGATIVE SIGN BITS
012422  060000                  PICST1: INTX!MINUSX
012424  001777                          MAXY
012426  041777                          INTX!MAXX
012430  020000                          MINUSY
012432  060000                          INTX!MINUSX
012434  021777                          MINUSX!MAXY
012436  061777                          INTX!MINUSX!MAXX
012440  020000                          MINUSY
                                ;NOW DRAW LOWER LEFT TO UPPER RIGHT DIAG.
012442  041777                          INTX!MAXX
```

```
  1480  012444  001777                          MAXY
  1481  012446  061777                          INTX!MINUSX!MAXX
  1482  012450  021777                          MINUSX!MAXY
  1483                                ;REPOSITION TO LOWER RIGHT AND DRAW LOWER RIGHT
  1484                                :         TO UPPER LEFT DIAG.
  1485  012452  001777                          MAXX
  1486  012454  000000                          0
  1487  012456  061777                          INTX!MINUSX!MAXX
  1488  012460  001777                          MAXY
  1489  012462  041777                          INTX!MAXX
  1490  012464  021777                          MINUSX!MAXY
  1491                                          .SBTTL  MENU 1 SUB-PICTURE
  1492                                ;DRAW REF. BOX IN THE MENU
  1493  012466  170003                          DMENU1                          ;ENABLE MENU
  1494  012470  114000                          POINT
  1495  012472  000000                          0
  1496  012474  000000                          0
  1497  012476  110000                          LONGV                           ;DRAW REF. BOX
  1498  012500  040177                          INTX!177
  1499  012502  000000                          0
  1500  012504  040000                          INTX
  1501  012506  001777                          MAXY
  1502  012510  060177                          INTX!MINUSX!177
  1503  012512  000000                          0
  1504  012514  040000                          INTX
  1505  012516  021777                          MINUSX!MAXY
  1506                                ;NOW REVERSE THE DRAWING PROCEDURE
  1507  012520  060000                          INTX!MINUSX
  1508  012522  001777                          MAXY
  1509  012524  040177                          INTX!177
  1510  012526  020000                          MINUSX
  1511  012530  060000                          INTX!MINUSX
  1512  012532  021777                          MINUSX!MAXY
  1513  012534  060177                          INTX!MINUSX!177
  1514  012536  020000                          MINUSX
  1515                                ;NOW DRAW THE DIAG. X IN THE MENU
  1516  012540  040177                          INTX!177                        ;LOWER LEFT, IN MENU, TO UPPER RIGHT
  1517  012542  001777                          MAXY
  1518  012544  060177                          INTX!MINUSX!177
  1519  012546  021777                          MINUSX!MAXY
  1520  012550  000177                          177                             ;REPOSITION TO LOWER LEFT OF MENU
  1521  012552  000000                          0
  1522  012554  060177                          INTX!MINUSX!177                 ;LOWER RIGHT TO UPPER LEFT
  1523  012556  001777                          MAXY
  1524  012560  040177                          INTX!177
  1525  012562  021777                          MINUSX!MAXY
  1526  012564  170002                          DMENU0                          ;RETURN TO MAIN SCREEN
  1527                                ;CONTINUE MAIN SCREEN PICTURE
  1528  012566  114000                          POINT
  1529  012570  001400                          1400
  1530  012572  001000                          1000
  1531                                ;DRAW A 100 UNIT BOX, SAME METHOD AS OUTER REF. BOX
  1532  012574  110000                          LONGV
  1533  012576  040144                          INTX!100.                       ; +X, +Y
```

```
1534  012600  000000                    0
1535  012602  040000                    INTX                            ; +X, +Y
1536  012604  000144                    100.
1537  012606  060144                    INTX!MINUSX!100.                ; -X, +Y
1538  012610  000000                    0
1539  012612  040000                    INTX                            ; +X, -Y
1540  012614  020144                    MINUSY!100.
1541  012616  040144                    INTX!100.                       ; +X, -Y
1542  012620  020000                    MINUSY
1543  012622  060000                    INTX!MINUSX                     ; -X, +Y
1544  012624  000144                    100.
1545  012626  060144                    INTX!MINUSX!100.                ; -X, -Y
1546  012630  020000                    MINUSY
1547  012632  060000                    INTX!MINUSX                     ; -X, -Y
1548  012634  020144                    MINUSY!100.
1549  012636  040144                    INTX!100.
1550  012640  000144                    100.
1551  012642  060144                    INTX!MINUSX!100.
1552  012644  020144                    MINUSX!100.
1553  012646  000144                    100.
1554  012650  000000                    0
1555  012652  060144                    INTX!MINUSX!100.
1556  012654  000144                    100.
1557  012656  040144                    INTX!100.
1558  012660  020144                    MINUSX!100.
1559                          ;DRAW A 10 UNIT BOX, SAME METHOD AS OUTER BOX
1560  012662  114000                    POINT
1561  012664  001400                    1400
1562  012666  000700                    700
1563  012670  110000                    LONGV
1564  012672  040012                    INTX!10.                        ; +X, +Y
1565  012674  000000                    0
1566  012676  040000                    INTX                            ; +X, +Y
1567  012700  000012                    10.
1568  012702  060012                    INTX!MINUSX!10.                 ; -X, +Y
1569  012704  000000                    0
1570  012706  040000                    INTX                            ; +X, -Y
1571  012710  020012                    MINUSY!10.
1572  012712  040012                    INTX!10.                        ; +X, -Y
1573  012714  020000                    MINUSY
1574  012716  060000                    INTX!MINUSX                     ; -X, +Y
1575  012720  000012                    10.
1576  012722  060012                    INTX!MINUSX!10.                 ; -X, -Y
1577  012724  020000                    MINUSY
1578  012726  060000                    INTX!MINUSX                     ; -X, -Y
1579  012730  020012                    MINUSY!10.
1580  012732  040012                    INTX!10.
1581  012734  000012                    10.
1582  012736  060012                    INTX!MINUSX!10.
1583  012740  020012                    MINUSX!10.
1584  012742  000012                    10.
1585  012744  000000                    0
1586  012746  060012                    INTX!MINUSX!10.
1587  012750  000012                    10.
```

```
  1588  012752  040012                          INTX!10.
  1589  012754  020012                          MINUSX!10.
  1590                                  ;DRAW FOUR VECTORS FROM A "COMMON" POINT WHICH WILL BE THE
  1591                                  ;SUPERIMPOSED UPON BY THE NEXT SUB-PICTURE
  1592  012756  114000                          POINT
  1593  012760  001000                          1000
  1594  012762  000400                          400
  1595  012764  110000                          LONGV
  1596  012766  040000                          INTX
  1597  012770  000200                          200
  1598  012772  114000                          POINT
  1599  012774  001000                          1000
  1600  012776  000400                          400
  1601  013000  110000                          LONGV
  1602  013002  040200                          INTX!200
  1603  013004  000000                          0
  1604  013006  114000                          POINT
  1605  013010  001000                          1000
  1606  013012  000400                          400
  1607  013014  110000                          LONGV
  1608  013016  040000                          INTX
  1609  013020  020200                          MINUSY!200
  1610  013022  114000                          POINT
  1611  013024  001000                          1000
  1612  013026  000400                          400
  1613  013030  110000                          LONGV
  1614  013032  060200                          INTX!MINUSX!200
  1615  013034  000000                          0
  1616                                          .SBTTL  DRAW 10 VERTICAL VECTORS IN THE LEFT CENTER AERA
  1617  013036  114000                          POINT
  1618  013040  000200                          200
  1619  013042  000740                          740
  1631  013044  104000                          SHORTV
   (1)  013046  040010                          INTX!10                 ;DRAW A 8. UNIT VERTICAL VECTOR
   (1)  013050  130000                          RELATP
   (1)  013052  040002                          INTX!2                  ;INTENSIFY A POINT 2 UNITS AWAY
   (1)  013054  000002                          2                       ;MOVE THE Y AXIS
   (1)  013056  104000                          SHORTV
   (1)  013060  060010                          INTX!MINUSX!10          ;DRAW A 8. UNIT VERTICAL VECTOR
   (1)  013062  130000                          RELATP
   (1)  013064  040002                          INTX!2                  ;INTENSIFY A POINT 2 UNITS AWAY
   (1)  013066  000002                          2                       ;MOVE THE Y AXIS
   (1)  013070  104000                          SHORTV
   (1)  013072  040010                          INTX!10                 ;DRAW A 8. UNIT VERTICAL VECTOR
   (1)  013074  130000                          RELATP
   (1)  013076  040002                          INTX!2                  ;INTENSIFY A POINT 2 UNITS AWAY
   (1)  013100  000002                          2                       ;MOVE THE Y AXIS
   (1)  013102  104000                          SHORTV
   (1)  013104  060010                          INTX!MINUSX!10          ;DRAW A 8. UNIT VERTICAL VECTOR
   (1)  013106  130000                          RELATP
   (1)  013110  040002                          INTX!2                  ;INTENSIFY A POINT 2 UNITS AWAY
   (1)  013112  000002                          2                       ;MOVE THE Y AXIS
   (1)  013114  104000                          SHORTV
   (1)  013116  040010                          INTX!10                 ;DRAW A 8. UNIT VERTICAL VECTOR
```

```
   (1)  013120  130000                    RELATP
   (1)  013122  040002                    INTX!2                          ;INTENSIFY A POINT 2 UNITS AWAY
   (1)  013124  000002                    2                               ;MOVE THE Y AXIS
   (1)  013126  104000                    SHORTV
   (1)  013130  060010                    INTX!MINUSX!10                  ;DRAW A 8. UNIT VERTICAL VECTOR
   (1)  013132  130000                    RELATP
   (1)  013134  040002                    INTX!2                          ;INTENSIFY A POINT 2 UNITS AWAY
   (1)  013136  000002                    2                               ;MOVE THE Y AXIS
   (1)  013140  104000                    SHORTV
   (1)  013142  040010                    INTX!10                         ;DRAW A 8. UNIT VERTICAL VECTOR
   (1)  013144  130000                    RELATP
   (1)  013146  040002                    INTX!2                          ;INTENSIFY A POINT 2 UNITS AWAY
   (1)  013150  000002                    2                               ;MOVE THE Y AXIS
   (1)  013152  104000                    SHORTV
   (1)  013154  060010                    INTX!MINUSX!10                  ;DRAW A 8. UNIT VERTICAL VECTOR
   (1)  013156  130000                    RELATP
   (1)  013160  040002                    INTX!2                          ;INTENSIFY A POINT 2 UNITS AWAY
   (1)  013162  000002                    2                               ;MOVE THE Y AXIS
   (1)  013164  104000                    SHORTV
   (1)  013166  040010                    INTX!10                         ;DRAW A 8. UNIT VERTICAL VECTOR
   (1)  013170  130000                    RELATP
   (1)  013172  040002                    INTX!2                          ;INTENSIFY A POINT 2 UNITS AWAY
   (1)  013174  000002                    2                               ;MOVE THE Y AXIS
   (1)  013176  104000                    SHORTV
   (1)  013200  060010                    INTX!MINUSX!10                  ;DRAW A 8. UNIT VERTICAL VECTOR
   (1)  013202  130000                    RELATP
   (1)  013204  040002                    INTX!2                          ;INTENSIFY A POINT 2 UNITS AWAY
   (1)  013206  000002                    2                               ;MOVE THE Y AXIS
1632                                      .SBTTL  DRAW THE DELAY INTENSITY SUB-PICTURE IN THE LEFT CENTER AERA
1633                               ;      DRAW 8 VECTORS USING BASIC VECTOR INSTRUCTION AWAY FROM A COMMON POINT
1634                               ;              BUT OFFSET BY ONE UNIT.
1635                               ;      THE COMMON POINT X=353 AND Y =1003
1636
1637             000353                   XQ6=353
1638             001003                   YQ6=1003
1639             000040                   LQ6=40
1640  013210  114000                      POINT
1641  013212  000354                      XQ6+1
1642  013214  001003                      YQ6
1643  013216  120000                      BASICV
1644  013220  042040                      INTX!PATH0!LQ6
1645  013222  114000                      POINT
1646  013224  000354                      XQ6+1
1647  013226  001004                      YQ6+1                           ;VECTOR #1
1648  013230  120000                      BASICV
1649  013232  046040                      INTX!PATH1!LQ6
1650  013234  114000                      POINT
1651  013236  000353                      XQ6
1652  013240  001004                      YQ6+1                           ;VECTOR #2
1653  013242  120000                      BASICV
1654  013244  052040                      INTX!PATH2!LQ6
1655  013246  114000                      POINT
1656  013250  000352                      XQ6-1
1657  013252  001004                      YQ6+1                           ;VECTOR #3
```

```
  1658  013254  120000                    BASICV
  1659  013256  056040                    INTX!PATH3!LQ6
  1660  013260  114000                    POINT
  1661  013262  000352                    XQ6-1
  1662  013264  001003                    YQ6                         ;VECTOR #4
  1663  013266  120000                    BASICV
  1664  013270  062040                    INTX!PATH4!LQ6
  1665  013272  114000                    POINT
  1666  013274  000352                    XQ6-1
  1667  013276  001002                    YQ6-1                       ;VECTOR #5
  1668  013300  120000                    BASICV
  1669  013302  066040                    INTX!PATH5!LQ6
  1670  013304  114000                    POINT
  1671  013306  000353                    XQ6
  1672  013310  001002                    YQ6-1                       ;VECTOR #6
  1673  013312  120000                    BASICV
  1674  013314  072040                    INTX!PATH6!LQ6
  1675  013316  114000                    POINT
  1676  013320  000354                    XQ6+1
  1677  013322  001002                    YQ6-1                       ;VECTOR #7
  1678  013324  120000                    BASICV
  1679  013326  076040                    INTX!PATH7!LQ6
  1680
  1681  013330  173400                    DSTOP
  1682  013332  160000                    DJMP
  1683  013334  012306                    FRME2
```

```
 1685                                   .SBTTL  OCTAGONS USING LONG AND ABSOLUTE VECTORS  (WIDTHS OF 7,77,177,377 AND 52
 1686                                   .SBTTL  CIRCLES USING ABSOLUTE VECTORS (WIDTHS OF 64., 128., AND 256.)
 1687
 1688  013336  114000          FRME3:   POINT
 1689  013340  000774                   774
 1690  013342  000764                   764
 1691  013344  164700                   CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
 1711  013346  110000                   LONGV                           ;OCTOGON BY LENGTH OF 7
  (1)  013350  040007                   INTX+7
  (1)  013352  000000                   0
  (1)  013354  040007                   INTX+7
  (1)  013356  000007                   7
  (1)  013360  040000                   INTX
  (1)  013362  000007                   7
  (1)  013364  060007                   INTX!MINUSX+7
  (1)  013366  000007                   7
  (1)  013370  060007                   INTX!MINUSX+7
  (1)  013372  000000                   0
  (1)  013374  060007                   INTX!MINUSX+7
  (1)  013376  020007                   MINUSX+7
  (1)  013400  040000                   INTX
  (1)  013402  020007                   MINUSX+7
  (1)  013404  040007                   INTX+7
  (1)  013406  020007                   MINUSX+7
```

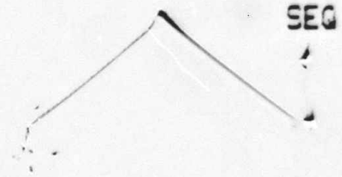
```
                                        ;CIRCLE 8 DEG. RADIUS OF 64

                                                POINT
                                                1077
                                                777
                                                ABSVCT                          ;ENABLE ABSOLUTE VECTOR MODE
1713
1714
1715  013410  114000                            .WORD   INTX!1076,1010
1716  013412  001077
1717  013414  000777
1718  013416  144000
1719  013420  041076  001010                    .WORD   INTX!1076,1010
1720  013424  041075  001021                    .WORD   INTX!1075,1021
1721  013430  041071  001031                    .WORD   INTX!1071,1031
1722  013434  041065  001041                    .WORD   INTX!1065,1041
1723  013440  041060  001050                    .WORD   INTX!1060,1050
1724  013444  041052  001057                    .WORD   INTX!1052,1057
1725  013450  041043  001064                    .WORD   INTX!1043,1064
1726  013454  041033  001071                    .WORD   INTX!1033,1071
1727  013460  041023  001074                    .WORD   INTX!1023,1074
1728  013464  041012  001076                    .WORD   INTX!1012,1076
1729  013470  041001  001077                    .WORD   INTX!1001,1077
1730  013474  040771  001077                    .WORD   INTX!771,1077
1731  013500  040761  001075                    .WORD   INTX!761,1075
1732  013504  040750  001072                    .WORD   INTX!750,1072
1733  013510  040740  001066                    .WORD   INTX!740,1066
1734  013514  040731  001061                    .WORD   INTX!731,1061
1735  013520  040722  001053                    .WORD   INTX!722,1053
1736  013524  040714  001045                    .WORD   INTX!714,1045
1737  013530  040707  001035                    .WORD   INTX!707,1035
1738  013534  040704  001025                    .WORD   INTX!704,1025
1739  013540  040701  001014                    .WORD   INTX!701,1014
1740  013544  040700  001003                    .WORD   INTX!700,1003
1741  013550  040700  000774                    .WORD   INTX!700,774
1742  013554  040701  000763                    .WORD   INTX!701,763
1743  013560  040704  000752                    .WORD   INTX!704,752
1744  013564  040707  000742                    .WORD   INTX!707,742
1745  013570  040714  000732                    .WORD   INTX!714,732
1746  013574  040722  000724                    .WORD   INTX!722,724
1747  013600  040731  000716                    .WORD   INTX!731,716
1748  013604  040740  000711                    .WORD   INTX!740,711
1749  013610  040750  000705                    .WORD   INTX!750,705
1750  013614  040760  000702                    .WORD   INTX!760,702
1751  013620  040771  000700                    .WORD   INTX!771,700
1752  013624  041001  000700                    .WORD   INTX!1001,700
1753  013630  041012  000701                    .WORD   INTX!1012,701
1754  013634  041023  000703                    .WORD   INTX!1023,703
1755  013640  041033  000706                    .WORD   INTX!1033,706
1756  013644  041043  000713                    .WORD   INTX!1043,713
1757  013650  041052  000720                    .WORD   INTX!1052,720
1758  013654  041060  000727                    .WORD   INTX!1060,727
1759  013660  041065  000736                    .WORD   INTX!1065,736
1760  013664  041071  000746                    .WORD   INTX!1071,746
1761  013670  041075  000756                    .WORD   INTX!1075,756
1762  013674  041076  000767                    .WORD   INTX!1076,767
1763  013700  041077  000777                    .WORD   INTX!1077,777
1764  013704  164000                            DNOP
1765  013706  164000                            DNOP
1766  013710  164000                            DNOP
```

```
 1767  013712  164000                          DNOP
 1768  013714  164000                          DNOP
 1769                              ;CIRCLE 8 DEG. RADIUS OF 128
 1770
 1771  013716  114000                          POINT
 1772  013720  001177                          1177
 1773  013722  000777                          777
 1774  013724  144000                          ABSVCT                          ;DISPLAY IN ABSOLUTE VECTOR MODE
 1775  013726  041176  001021                  .WORD   INTX!1176,1021
 1776  013732  041172  001042                  .WORD   INTX!1172,1042
 1777  013736  041164  001063                  .WORD   INTX!1164,1063
 1778  013742  041154  001103                  .WORD   INTX!1154,1103
 1779  013746  041141  001121                  .WORD   INTX!1141,1121
 1780  013752  041125  001136                  .WORD   INTX!1125,1136
 1781  013756  041107  001151                  .WORD   INTX!1107,1151
 1782  013762  041067  001162                  .WORD   INTX!1067,1162
 1783  013766  041047  001171                  .WORD   INTX!1047,1171
 1784  013772  041025  001175                  .WORD   INTX!1025,1175
 1785  013776  041003  001177                  .WORD   INTX!1003,1177
 1786  014002  040763  001176                  .WORD   INTX!763,1176
 1787  014006  040741  001173                  .WORD   INTX!741,1173
 1788  014012  040720  001166                  .WORD   INTX!720,1166
 1789  014016  040700  001156                  .WORD   INTX!700,1156
 1790  014022  040661  001144                  .WORD   INTX!661,1144
 1791  014026  040644  001130                  .WORD   INTX!644,1130
 1792  014032  040630  001112                  .WORD   INTX!630,1112
 1793  014036  040617  001073                  .WORD   INTX!617,1073
 1794  014042  040610  001053                  .WORD   INTX!610,1053
 1795  014046  040603  001032                  .WORD   INTX!603,1032
 1796  014052  040600  001010                  .WORD   INTX!600,1010
 1797  014056  040600  000767                  .WORD   INTX!600,767
 1798  014062  040603  000745                  .WORD   INTX!603,745
 1799  014066  040610  000724                  .WORD   INTX!610,724
 1800  014072  040617  000704                  .WORD   INTX!617,704
 1801  014076  040630  000665                  .WORD   INTX!630,665
 1802  014102  040644  000647                  .WORD   INTX!644,647
 1803  014106  040661  000633                  .WORD   INTX!661,633
 1804  014112  040700  000621                  .WORD   INTX!700,621
 1805  014116  040720  000611                  .WORD   INTX!720,611
 1806  014122  040741  000604                  .WORD   INTX!741,604
 1807  014126  040763  000601                  .WORD   INTX!763,601
 1808  014132  041003  000600                  .WORD   INTX!1003,600
 1809  014136  041025  000602                  .WORD   INTX!1025,602
 1810  014142  041047  000606                  .WORD   INTX!1047,606
 1811  014146  041067  000615                  .WORD   INTX!1067,615
 1812  014152  041107  000626                  .WORD   INTX!1107,626
 1813  014156  041125  000641                  .WORD   INTX!1125,641
 1814  014162  041141  000656                  .WORD   INTX!1141,656
 1815  014166  041154  000674                  .WORD   INTX!1154,674
 1816  014172  041164  000714                  .WORD   INTX!1164,714
 1817  014176  041172  000735                  .WORD   INTX!1172,735
 1818  014202  041176  000756                  .WORD   INTX!1176,756
 1819  014206  041177  000777                  .WORD   INTX!1177,777
 1820  014212  164000                          DNOP
```

```
                                        DNOP
                                        DNOP
                                        DNOP
                                ;CIRCLE 8 DEG. RADIUS OF 256

                                        POINT
                                        1377
                                        777
                                        ABSVCT                          ;ENABLE ABSOLUTE VECTOR MODE
                                        .WORD   INTX!1375,1043
                                        .WORD   INTX!1365,1106
                                        .WORD   INTX!1351,1147
                                        .WORD   INTX!1330,1207
                                        .WORD   INTX!1303,1244
                                        .WORD   INTX!1252,1275
                                        .WORD   INTX!1216,1323
                                        .WORD   INTX!1157,1345
                                        .WORD   INTX!1116,1362
                                        .WORD   INTX!1053,1373
                                        .WORD   INTX!1010,1377
                                        .WORD   INTX!745,1376
                                        .WORD   INTX!702,1367
                                        .WORD   INTX!640,1354
                                        .WORD   INTX!600,1335
                                        .WORD   INTX!542,1311
                                        .WORD   INTX!510,1261
                                        .WORD   INTX!461,1226
                                        .WORD   INTX!436,1167
                                        .WORD   INTX!417,1127
                                        .WORD   INTX!406,1064
                                        .WORD   INTX!401,1021
                                        .WORD   INTX!401,755
                                        .WORD   INTX!406,713
                                        .WORD   INTX!417,651
                                        .WORD   INTX!436,610
                                        .WORD   INTX!461,552
                                        .WORD   INTX!510,516
                                        .WORD   INTX!542,466
                                        .WORD   INTX!600,442
                                        .WORD   INTX!640,423
                                        .WORD   INTX!702,410
                                        .WORD   INTX!745,401
                                        .WORD   INTX!1010,400
                                        .WORD   INTX!1053,404
                                        .WORD   INTX!1116,415
                                        .WORD   INTX!1157,432
                                        .WORD   INTX!1216,454
                                        .WORD   INTX!1252,502
                                        .WORD   INTX!1303,533
                                        .WORD   INTX!1330,570
                                        .WORD   INTX!1351,630
                                        .WORD   INTX!1365,671
                                        .WORD   INTX!1374,734
                                        .WORD   INTX!1377,777
```



```
 1875  014516  164000                 DNOP
 1876  014520  164000                 DNOP
 1877  014522  164000                 DNOP
 1878  014524  164000                 DNOP
 1879  014526  114000                 POINT
 1880  014530  000740                 740
 1881  014532  000640                 640
 1882  014534  110000                 LONGV                    ;OCTOGON BY LENGTH OF 77
   (1) 014536  040077                 INTX+77
   (1) 014540  000000                 0
   (1) 014542  040077                 INTX+77
   (1) 014544  000077                 77
   (1) 014546  040000                 INTX
   (1) 014550  000077                 77
   (1) 014552  060077                 INTX!MINUSX+77
   (1) 014554  000077                 77
   (1) 014556  060077                 INTX!MINUSX+77
   (1) 014560  000000                 0
   (1) 014562  060077                 INTX!MINUSX+77
   (1) 014564  020077                 MINUSX+77
   (1) 014566  040000                 INTX
   (1) 014570  020077                 MINUSX+77
   (1) 014572  040077                 INTX+77
   (1) 014574  020077                 MINUSX+77
 1883  014576  114000                 POINT
 1884  014600  000700                 700
 1885  014602  000500                 500
 1886  014604  110000                 LONGV                    ;OCTOGON BY LENGTH OF 177
   (1) 014606  040177                 INTX+177
   (1) 014610  000000                 0
   (1) 014612  040177                 INTX+177
   (1) 014614  000177                 177
   (1) 014616  040000                 INTX
   (1) 014620  000177                 177
   (1) 014622  060177                 INTX!MINUSX+177
   (1) 014624  000177                 177
   (1) 014626  060177                 INTX!MINUSX+177
   (1) 014630  000000                 0
   (1) 014632  060177                 INTX!MINUSX+177
   (1) 014634  020177                 MINUSX+177
   (1) 014636  040000                 INTX
   (1) 014640  020177                 MINUSX+177
   (1) 014642  040177                 INTX+177
   (1) 014644  020177                 MINUSX+177
 1887  014646  114000                 POINT
 1888  014650  000600                 600
 1889  014652  000200                 200
 1890  014654  110000                 LONGV                    ;OCTOGON BY LENGTH OF 377
   (1) 014656  040377                 INTX+377
   (1) 014660  000000                 0
   (1) 014662  040377                 INTX+377
   (1) 014664  000377                 377
   (1) 014666  040000                 INTX
   (1) 014670  000377                 377
```

```
   (1)  014672  060377                    INTX!MINUSX+377
   (1)  014674  000377                    377
   (1)  014676  060377                    INTX!MINUSX+377
   (1)  014700  000000                    0
   (1)  014702  060377                    INTX!MINUSX+377
   (1)  014704  020377                    MINUSX+377
   (1)  014706  040000                    INTX
   (1)  014710  020377                    MINUSX+377
   (1)  014712  040377                    INTX+377
   (1)  014714  020377                    MINUSX+377
1891                              ;DRAW ABSOLUTE VECTOR OCTAGON
1892    014716  114000                    POINT
1893    014720  000530                    530
1894    014722  000010                    10
1895    014724  144000                    ABSVCT
1896    014726  041250                    INTX!1250                       ; #1
1897    014730  000010                    10
1898    014732  041770                    INTX!1770                       ; #2
1899    014734  000530                    530
1900    014736  041770                    INTX!1770                       ; #3
1901    014740  001250                    1250
1902    014742  041250                    INTX!1250                       ; #4
1903    014744  001770                    1770
1904    014746  040530                    INTX!530                        ; #5
1905    014750  001770                    1770
1906    014752  040010                    INTX!10                         ; #6
1907    014754  001250                    1250
1908    014756  040010                    INTX!10                         ; #7
1909    014760  000530                    530
1910    014762  040530                    INTX!530                        ; #8
1911    014764  000010                    10
1912                              ;DRAW A BASIC SHORT VECTOR OCTAGON
1913    014766  114000                    POINT
1914    014770  000300                    300
1915    014772  001000                    1000
1916    014774  134000                    BASICS                          ;BASIC SHORT VECTOR
1917    014776  073777                    73777                           ;PATH 6 & 7
1918    015000  063737                    63737                           ;PATH 4 & 5
1919    015002  053677                    53677                           ;PATH 2 & 3
1920    015004  043637                    43637                           ;PATH 0 & 1
1921                              ;DRAW ANOTHER IN THE RIGHT CENTER
1922    015006  114000                    POINT
1923    015010  001500                    1500
1924    015012  001000                    1000
1925    015014  134000                    BASICS
1926    015016  073737                    73737                           ;PATH 6 & 5
1927    015020  043777                    43777                           ;PATH 0 & 7
1928    015022  053637                    53637                           ;PATH 2 & 1
1929    015024  063677                    63677                           ;PATH 4 & 3
1930    015026  173400                    DSTOP
1931    015030  160000                    DJMP
1932    015032  013336                    FRME3
1933                                      .SBTTL  X AND Y OFFSET SUB-PICTURE
```

```
 1935  015034  114000                OFFTST: POINT
 1936  015036  010000                OFFT1:  BIT12
 1937  015040  010000                OFFT2:  BIT12
 1938  015042  164700                        CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
 1939  015044  116200                        POINT!INT1
 1940  015046  000400                        400
 1941  015050  000400                        400
 1942  015052  120000                        BASICV
 1943  015054  043000                        INTX!PATH0!1000                 ;DRAW A SQUARE
 1944  015056  053000                        INTX!PATH2!1000
 1945  015060  063000                        INTX!PATH4!1000
 1946  015062  073000                        INTX!PATH6!1000
 1947  015064  173400                        DSTOP
 1948  015066  160000                        DJMP
 1949  015070  015034                        OFFTST
 1950
 1951                                        .SBTTL  SUPER AND SUBSCRIPT SUB-PICTURE
 1952
 1953  015072  114000                SUPPIC: POINT
 1954  015074  000400                        400
 1955  015076  001000                        1000
 1956  015100  110000                        LONGV
 1957  015102  041000                        INTX!1000                       ;DRAW REF. LINE
 1958  015104  000000                        0
 1959  015106  114000                        POINT
 1960  015110  000400                        400
 1961  015112  001000                        1000
 1962  015114  160000                        DJMP                            ;BYPASS ROTATED REF. LINE
 1963  015116  015142                        SUPC1
 1964  015120  114000                SUPC0:  POINT
 1965  015122  001000                        1000
 1966  015124  000400                        400
 1967  015126  110000                        LONGV
 1968  015130  040000                        INTX
 1969  015132  001000                        1000
 1970  015134  114000                        POINT
 1971  015136  001000                        1000
 1972  015140  000400                        400
 1973  015142  154340                SUPC1:  CHARS3                          ;ENSURE MAX CHAR SIZE
 1974  015144  170040                        STATSA!ITALO
 1975  015146  100000                        CHAR
 1976  015150  162000                        DJSR
 1977  015152  015172                        SUPSUB
 1978  015154  170060                        STATSA!ITAL1                    ;SET ITALIC
 1979  015156  162000                        DJSR
 1980  015160  015172                        SUPSUB
 1981  015162  154240                        CHARS1
 1982  015164  173400                        DSTOP
 1983  015166  160000                        DJMP
 1984  015170  015072                        SUPPIC
 1985
 1986                                        .SBTTL  SUPER AND SUBSCRIPT ASCII STRING
 1987
 1988  015172     105     021     105 SUPSUB: .BYTE   105,SUPON,105,SUPON,105,SUPON,105
```

```
        015175      021     105     021
        015200      105
 1989                                           ;NOW REVERSE AND INCREASE SIZE
 1990   015201      023     105     023         .BYTE   SUPOFF,105,SUPOFF,105,SUPOFF,105
        015204      105     023     105
 1991                                   ;NOW IT SHOULD BE AT THE BIGGEST SIZE
 1992   015207      022     105     022         .BYTE   SUBON,105,SUBON,105,SUBON,105
        015212      105     022     105
 1993                                   ;REVERSE AND INCREASE SIZE
 1994   015215      024     105     024         .BYTE   SUBOFF,105,SUBOFF,105,SUBOFF,105
        015220      105     024     105
 1995   015223      040                         .BYTE   40
 1996
 1997   015224  166000                        DPOP
 1998
 1999
 2000                                           .SBTTL  SYNC SPEED SUBPICTURE
 2001
 2002   015226  170000                SYNPIC: STATSA                          ;VARIABLE WORD TO HANDLE SYNC SPEED
 2003   015230  164700                        CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
 2004   015232  114000                        POINT
 2005   015234  001000                        1000
 2006   015236  000000                        0
 2007   015240  120000                        BASICV
 2008   015242  047000                        INTX!PATH1!1000                 ;DRAW A DIAMOND
 2009   015244  057000                        INTX!PATH3!1000
 2010   015246  067000                        INTX!PATH5!1000
 2011   015250  077000                        INTX!PATH7!1000
 2012   015252  114000                        POINT
 2013   015254  000600                        600
 2014   015256  001000                        1000
 2015   015260  176003                        STRNG1                          ;ENABLE CHARACTER TERMINATE
 2016   015262  100000                        CHAR
 2017   015264  162000                        DJSR
 2018   015266  015300                        SYNTXT                          ;DISPLAY SYNC SPEED MESSAGE
 2019   015270  176002                        STRNGO                          ;DISABLE CHARACTER STRING ESCAPE
 2020   015272  173400                        DSTOP
 2021   015274  160000                        DJMP
 2022   015276  015226                        SYNPIC                          ;CONTINUE
 2023
 2024   015300  044124  051511  043040 SYNTXT: .ASCII  /THIS FRAME USES /
        015306  040522  042515  052440
        015314  042523  020123
 2025   015320  047516                SYNSPD: .ASCII  /NO/
 2026   015322  051440  047131  020103        .ASCII  / SYNC              /
        015330  020040  020040  020040
        015336  020040  020040  020040
        015344     040
 2027   015345     177                        .BYTE   177
 2028
 2029                                   ;SHOULD NEVER GET HERE UNLESS CHAR TERM. FAILS
 2030   015346  114000                1$:     POINT
 2031   015350  000200                        200
 2032   015352  000700                        700
```

```
                                        CHAR
                                        .ASCIZ  /CHARACTER TERMINATE FAILED TO CAUSE A POP AND RESTORE/
                                        DJMP
                                        I$

                                        .SBTTL  DASH LINE SUB-PICTURE

                                FRME5:  POINT!INT4
                                        0
                                        1000
                                        CHARS1
                                        CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
                                        CHAR!LINE0
                                        .ASCII /SOLID     /
                                        LONGV!LINE0
                                        40400
                                        0
                                        400
                                        0
                                        LONGV!BLKON
                                        40400
                                        0
                                        CHAR!BLKOFF
                                        .BYTE   15,12,12,12,12,12
                                        .ASCII /DASH I    /
                                        LONGV!LINE1
                                        40400
                                        0
                                        400
                                        0
                                        LONGV!BLKON
                                        40400
                                        0
                                        CHAR!BLKOFF
                                        .BYTE   15,12,12,12,12,12
                                        .ASCII /DASH II   /
                                        LONGV!LINE2
                                        40400
                                        0
                                        400
                                        0
```

```
  2074  015614  110030                          LONGV!BLKON
  2075  015616  040400                          40400
  2076  015620  000000                          0
  2077  015622  100020                          CHAR!BLKOFF
  2078  015624     015     012     012          .BYTE   15,12,12,12,12,12
        015627     012     012     012
  2079  015632  040504  044123  044440          .ASCII /DASH III  /
        015640  044511  020040
  2080  015644  110007                          LONGV!LINE3
  2081  015646  040400                          40400
  2082  015650  000000                          0
  2083  015652  000400                          400
  2084  015654  000000                          0
  2085  015656  110030                          LONGV!BLKON
  2086  015660  040400                          40400
  2087  015662  000000                          0
  2088  015664  110024                          LONGV!BLKOFF!LINE0
  2089  015666  000000                          0
  2090  015670  000000                          0
  2091  015672  173400                          DSTOP
  2100  015674  161665                          DJMPR!BIT8!WHERE1               ;DJMP RELATIVE TO THE TAG "FRME5"
  2101
  2102                                          .SBTTL  VECTOR LENGTH SUB-PICTURE
  2103
  2104  015676  154024              FRME6:      VCTR00!4                        ;NORMAL VECTOR
  2105  015700  114000                          POINT
  2106  015702  001777                          MAXX
  2107  015704  000000                          0
  2108  015706  164700                          CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
  2109  015710  113600                          LONGV!INT7
  2110  015712  040000                          INTX
  2111  015714  001777                          MAXY
  2112  015716  114000                          POINT
  2113  015720  000000                          0
  2114  015722  001777                          MAXY
  2115  015724  110000                          LONGV
  2116  015726  041777                          INTX!MAXX
  2117  015730  000000                          0
  2118  015732  114000                          POINT
  2119  015734  000000                          0
  2120  015736  000000                          0
  2121  015740  154037                          VCTR00!17                       ;MAX LENGTH VECTOR
  2122  015742  144000                          ABSVCT                          ;ABSOLUTE VECTOR
  2123  015744  160000                          DJMP
  2124  015746  026070                          BUFFER
  2125
  2126
  2127                                          .SBTTL  HORIZONTAL PHOSPHOR SUB-PICTURE
  2128
```

```
 2130
 2131  015750  114000  FRME10: POINT
 2132  015752  000000  DELTX7: 0
 2133  015754  000000         0
 2134  015756  123600  DFI10A: BASICV!INT7
 2140  015760  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  015762  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  015764  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  015766  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  015770  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  015772  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  015774  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  015776  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016000  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  016002  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016004  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  016006  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016010  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  016012  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016014  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  016016  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016020  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  016022  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016024  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  016026  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016030  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  016032  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016034  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  016036  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016040  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  016042  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016044  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  016046  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016050  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  016052  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016054  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  016056  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016060  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  016062  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016064  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  016066  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016070  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  016072  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016074  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  016076  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016100  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  016102  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016104  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  016106  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016110  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
  (1)  016112  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016114  073777          INTX!PATH6!MAXY              ;VECTOR DOWN
  (1)  016116  002002          PATH0!2                      ;MOVE RIGHT 2 UNITS
  (1)  016120  053777          INTX!PATH2!MAXY              ;VECTOR STRAIGHT UP
```

```
   (1)  016122  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016124  073777                INTX!PATH6!MAXY                 ;VECTOR DOWN
   (1)  016126  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016130  053777                INTX!PATH2!MAXY                 ;VECTOR STRAIGHT UP
   (1)  016132  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016134  073777                INTX!PATH6!MAXY                 ;VECTOR DOWN
   (1)  016136  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016140  053777                INTX!PATH2!MAXY                 ;VECTOR STRAIGHT UP
   (1)  016142  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016144  073777                INTX!PATH6!MAXY                 ;VECTOR DOWN
   (1)  016146  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016150  053777                INTX!PATH2!MAXY                 ;VECTOR STRAIGHT UP
   (1)  016152  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016154  073777                INTX!PATH6!MAXY                 ;VECTOR DOWN
   (1)  016156  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016160  053777                INTX!PATH2!MAXY                 ;VECTOR STRAIGHT UP
   (1)  016162  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016164  073777                INTX!PATH6!MAXY                 ;VECTOR DOWN
   (1)  016166  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016170  053777                INTX!PATH2!MAXY                 ;VECTOR STRAIGHT UP
   (1)  016172  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016174  073777                INTX!PATH6!MAXY                 ;VECTOR DOWN
   (1)  016176  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016200  053777                INTX!PATH2!MAXY                 ;VECTOR STRAIGHT UP
   (1)  016202  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016204  073777                INTX!PATH6!MAXY                 ;VECTOR DOWN
   (1)  016206  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016210  053777                INTX!PATH2!MAXY                 ;VECTOR STRAIGHT UP
   (1)  016212  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
   (1)  016214  073777                INTX!PATH6!MAXY                 ;VECTOR DOWN
   (1)  016216  002002                PATH0!2                         ;MOVE RIGHT 2 UNITS
2141    016220  173400                DSTOP
2142    016222  160000                DJMP
2143    016224  015756                DFI10A
2144
2145                                   .SBTTL  MAIN VERTICAL PHOSPHOR SUB-PICTURE
2146
2147    016226  114000        FRME11: POINT
2148    016230  000000                0
2149    016232  000000        DELTY7: 0
2150    016234  170002                DMENU0
2151    016236  123600        DFI11C: BASICV!INT7
2157    016240  043777                INTX!PATH0!MAXX                 ;VECTOR RIGHT FULL WIDTH
   (1)  016242  012002                PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016244  063777                INTX!PATH4!MAXX                 ;VECTOR LEFT FULL SCREEN
   (1)  016246  012002                PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016250  043777                INTX!PATH0!MAXX                 ;VECTOR RIGHT FULL WIDTH
   (1)  016252  012002                PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016254  063777                INTX!PATH4!MAXX                 ;VECTOR LEFT FULL SCREEN
   (1)  016256  012002                PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016260  043777                INTX!PATH0!MAXX                 ;VECTOR RIGHT FULL WIDTH
   (1)  016262  012002                PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016264  063777                INTX!PATH4!MAXX                 ;VECTOR LEFT FULL SCREEN
   (1)  016266  012002                PATH2!2                         ;MOVE UP 2 UNITS
```

```
(1)  016270  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016272  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016274  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016276  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016300  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016302  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016304  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016306  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016310  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016312  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016314  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016316  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016320  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016322  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016324  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016326  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016330  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016332  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016334  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016336  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016340  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016342  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016344  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016346  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016350  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016352  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016354  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016356  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016360  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016362  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016364  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016366  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016370  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016372  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016374  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016376  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016400  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016402  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016404  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016406  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016410  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016412  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016414  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016416  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016420  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016422  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016424  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016426  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016430  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016432  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016434  063777          INTX!PATH4!MAXX          ;VECTOR LEFT FULL SCREEN
(1)  016436  012002          PATH2!2                  ;MOVE UP 2 UNITS
(1)  016440  043777          INTX!PATH0!MAXX          ;VECTOR RIGHT FULL WIDTH
(1)  016442  012002          PATH2!2                  ;MOVE UP 2 UNITS
```

```
   (1)  016444  063777                    INTX!PATH4!MAXX                 ;VECTOR LEFT FULL SCREEN
   (1)  016446  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016450  043777                    INTX!PATH0!MAXX                 ;VECTOR RIGHT FULL WIDTH
   (1)  016452  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016454  063777                    INTX!PATH4!MAXX                 ;VECTOR LEFT FULL SCREEN
   (1)  016456  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016460  043777                    INTX!PATH0!MAXX                 ;VECTOR RIGHT FULL WIDTH
   (1)  016462  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016464  063777                    INTX!PATH4!MAXX                 ;VECTOR LEFT FULL SCREEN
   (1)  016466  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016470  043777                    INTX!PATH0!MAXX                 ;VECTOR RIGHT FULL WIDTH
   (1)  016472  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016474  063777                    INTX!PATH4!MAXX                 ;VECTOR LEFT FULL SCREEN
   (1)  016476  012002                    PATH2!2                         ;MOVE UP 2 UNITS
 2158   016500  173400                    DSTOP
 2159   016502  160000                    DJMP
 2160   016504  016236                    DFI11C
 2161
 2162                                     .SBTTL  MENU VERTICAL PHOSPHOR SUB-PICTURE
 2163
 2164   016506  114000            FRM11S: POINT
 2165   016510  000000                    0
 2166   016512  000000            DELT11: 0
 2167   016514  123600            FRM11D: BASICV!INT7
 2173   016516  042177                    INTX!PATH0!MAXMUX               ;VECTOR RIGHT FULL SCREEN IN MENU
   (1)  016520  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016522  062177                    INTX!PATH4!MAXMUX               ;VECTOR LEFT FULL MENU SCREEN
   (1)  016524  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016526  042177                    INTX!PATH0!MAXMUX               ;VECTOR RIGHT FULL SCREEN IN MENU
   (1)  016530  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016532  062177                    INTX!PATH4!MAXMUX               ;VECTOR LEFT FULL MENU SCREEN
   (1)  016534  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016536  042177                    INTX!PATH0!MAXMUX               ;VECTOR RIGHT FULL SCREEN IN MENU
   (1)  016540  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016542  062177                    INTX!PATH4!MAXMUX               ;VECTOR LEFT FULL MENU SCREEN
   (1)  016544  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016546  042177                    INTX!PATH0!MAXMUX               ;VECTOR RIGHT FULL SCREEN IN MENU
   (1)  016550  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016552  062177                    INTX!PATH4!MAXMUX               ;VECTOR LEFT FULL MENU SCREEN
   (1)  016554  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016556  042177                    INTX!PATH0!MAXMUX               ;VECTOR RIGHT FULL SCREEN IN MENU
   (1)  016560  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016562  062177                    INTX!PATH4!MAXMUX               ;VECTOR LEFT FULL MENU SCREEN
   (1)  016564  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016566  042177                    INTX!PATH0!MAXMUX               ;VECTOR RIGHT FULL SCREEN IN MENU
   (1)  016570  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016572  062177                    INTX!PATH4!MAXMUX               ;VECTOR LEFT FULL MENU SCREEN
   (1)  016574  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016576  042177                    INTX!PATH0!MAXMUX               ;VECTOR RIGHT FULL SCREEN IN MENU
   (1)  016600  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016602  062177                    INTX!PATH4!MAXMUX               ;VECTOR LEFT FULL MENU SCREEN
   (1)  016604  012002                    PATH2!2                         ;MOVE UP 2 UNITS
   (1)  016606  042177                    INTX!PATH0!MAXMUX               ;VECTOR RIGHT FULL SCREEN IN MENU
   (1)  016610  012002                    PATH2!2                         ;MOVE UP 2 UNITS
```

```
    (1)* 016612  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016614  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016616  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016620  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016622  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016624  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016626  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016630  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016632  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016634  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016636  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016640  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016642  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016644  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016646  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016650  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016652  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016654  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016656  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016660  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016662  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016664  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016666  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016670  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016672  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016674  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016676  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016700  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016702  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016704  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016706  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016710  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016712  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016714  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016716  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016720  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016722  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016724  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016726  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016730  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016732  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016734  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016736  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016740  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016742  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016744  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016746  042177          INTX!PATH0!MAXMUX          ;VECTOR RIGHT FULL SCREEN IN MENU
    (1)  016750  012002          PATH2!2                    ;MOVE UP 2 UNITS
    (1)  016752  062177          INTX!PATH4!MAXMUX          ;VECTOR LEFT FULL MENU SCREEN
    (1)  016754  012002          PATH2!2                    ;MOVE UP 2 UNITS
2174     016756  173400          DSTOP
2175     016760  160000          DJMP
2176     016762  016514          FRM11D
2177
```

```
016764 117600          FRM10:  POINT!INT7
016766 000000                  0
016770 000000                  0
016772 164700                  CONSL1!BIT7!BIT6          ;ENABLE CONSOLE #1
016774 110000                  LONGV
016776 041777                  INTX!MAXX
017000 000000                  0
017002 040000                  INTX
017004 001777                  MAXY
017006 061777                  INTX!MINUSX!MAXX
017010 000000                  0
017012 040000                  INTX
017014 021777                  MINUSX!MAXY
017016 173400                  DSTOP
017020 160000                  DJMP
017022 016764                  FRM10

017024 170003          FRM11M: DMENU1                    ;ENABLE MENU
017026 117600                  POINT!INT7
017030 000000                  0
017032 000000                  0
017034 110000                  LONGV
017036 040177                  INTX!MAXMUX
017040 000000                  0
017042 040000                  INTX
017044 001777                  MAXY
017046 060177                  INTX!MINUSX!MAXMUX
017050 000000                  0
017052 040000                  INTX
017054 021777                  MINUSX!MAXY
017056 173400                  DSTOP
017060 160000                  DJMP
017062 017024                  FRM11M
```

```
                                        .SBTTL  SHORT VECTOR AND RELATIVE POINT SUB-PICTURE


017064  164700                FRME14: CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
017066  114000                        POINT
017070  000000                FRM14A: 0
017072  000000                FRM14B: 0
017074  104000                        SHORTV
017076  056200                        INTX+16200
(1) 017100  056271                  INTX+16200+71
(1) 017102  040071                  INTX+71
(1) 017104  076271                  INTX!MINUSX+16200+71
(1) 017106  076200                  INTX!MINUSX+16200
(1) 017110  076371                  INTX!MINUSX+16200+MINSUY+71
(1) 017112  040171                  INTX+MINSUY+71
(1) 017114  056371                  INTX+16200+MINSUY+71
(1) 017116  020504                  20504
017120  130000                      RELATP
017122  057000                      INTX+17000
(1) 017124  057074                  INTX+17000+74
(1) 017126  040074                  INTX+74
(1) 017130  077074                  INTX!MINUSX+17000+74
(1) 017132  077000                  INTX!MINUSX+17000
(1) 017134  077174                  INTX!MINUSX+17000+MINSUY+74
(1) 017136  040174                  INTX+MINSUY+74
(1) 017140  057174                  INTX+17000+MINSUY+74
(1) 017142  020504                  20504
017144  104000                      SHORTV
017146  057600                      INTX+17600
(1) 017150  057677                  INTX+17600+77
(1) 017152  040077                  INTX+77
(1) 017154  077677                  INTX!MINUSX+17600+77
(1) 017156  077600                  INTX!MINUSX+17600
(1) 017160  077777                  INTX!MINUSX+17600+MINSUY+77
(1) 017162  040177                  INTX+MINSUY+77
(1) 017164  057777                  INTX+17600+MINSUY+77
(1) 017166  020504                  20504
017170  173400                      DSTOP
017172  160000                      DJMP
017174  017064                      FRME14
```

```
                                .SBTTL  GRAPHPLOT INCREMENT SUB-PICTURE

017176  114004          GRAPH:  POINT!LINE0
017200  000400                  400
017202  000200                  200
017204  164700                  CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
017206  110000                  LONGV                           ;DRAW BASE REF. VECTOR FOR GRAPH Y
017210  041200                  INTX+1200
017212  000000                  0

017214  114000                  POINT
017216  000440                  440
017220  000200                  200
017222  174104          GRPINC: STATSB!INCR+4                   ;LOAD GRAPHPLOT INCR. REGISTER
017224  124000                  GRAPHY
017226  162000                  DJSR                            ;DJSR TO "SINE DATA"
017230  017300                  SINE
017232  162000                  DJSR                            ;DJSR TO SINE DATE
017234  017300                  SINE

017236  114000                  POINT
017240  000200                  200
017242  000040                  40
017244  110000                  LONGV                           ;DRAW BASE REF. VECTOR FOR GRAPH X
017246  040000                  INTX
017250  001200                  1200

017252  114000                  POINT
017254  000200                  200
017256  000100                  100
017260  120000                  GRAPHX
017262  162000                  DJSR                            ;DJSR TO "SINE DATA"
017264  017300                  SINE
017266  162000                  DJSR                            ;DJSR TO "SINE DATA"
017270  017300                  SINE
017272  173400                  DSTOP
017274  160000                  DJMP
017276  017176                  GRAPH
```

```
2282
2283                                                    .SBTTL  DATA STRING FOR A SINE WAVE
2284
2285  017300  000200  000205  000212  SINE:   .WORD   0200,0205,0212,0217,0224,0231,0236,0243,0247,0253
      017306  000217  000224  000231
      017314  000236  000243  000247
      017322  000253
2286  017324  000257  000262  000265          .WORD   0257,0262,0265,0270,0272,0274,0276,0277,0277,0277
      017332  000270  000272  000274
      017340  000276  000277  000277
      017346  000277
2287  017350  000277  000276  000275          .WORD   0277,0276,0275,0274,0272,0267,0264,0261,0256,0252
      017356  000274  000272  000267
      017364  000264  000261  000256
      017372  000252
2288  017374  000246  000241  000235          .WORD   0246,0241,0235,0230,0223,0216,0211,0203,0176,0171
      017402  000230  000223  000216
      017410  000211  000203  000176
      017416  000171
2289  017420  000163  000156  000151          .WORD   0163,0156,0151,0144,0137,0133,0127,0123,0117,0114
      017426  000144  000137  000133
      017434  000127  000123  000117
      017442  000114
2290  017444  000111  000106  000104          .WORD   0111,0106,0104,0102,0101,0100,0100,0100,0100,0101
      017452  000102  000101  000100
      017460  000100  000100  000100
      017466  000101
2291  017470  000102  000104  000106          .WORD   0102,0104,0106,0111,0113,0117,0122,0126,0132,0137
      017476  000111  000113  000117
      017504  000122  000126  000132
      017512  000137
2292  017514  000144  000151  000156          .WORD   0144,0151,0156,0163,0170,0175
      017522  000163  000170  000175
2293
2294  017530  166000                          DPOP                            ;DISPLAY POP AND RESTORE
2295
2296                                          .SBTTL  SHORT TERM DRIFT SUB-PICTURE
```

```
  2298  017532  164700                STDPIC: CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
  2299  017534  117000                        POINT!INT4
  2300  017536  000000                STDRA:  0
  2301  017540  000000                STDRB:  0
  2302  017542  173400                        DSTOP
  2303  017544  160000                        DJMP
  2304  017546  017532                        STDPIC
  2305
  2306  017550  114000                PICVTL: POINT
  2307  017552  000000                PICVTA: 0
  2308  017554  000000                PICVTB: 0
  2309  017556  110000                        LONGV
  2310  017560  040000                PICVTC: INTX
  2311  017562  000000                        0
  2312  017564  173400                PICVTE: DSTOP
  2313  017566  160000                        DJMP
  2314  017570  017550                        PICVTL
  2315
  2316                                  .SBTTL  SCREEN SCISSORING SUB-PICTURE
  2317
  2318  017572  164700                PICSCS: CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
  2319  017574  154024                        VCTR00!4
  2320  017576  114000                        POINT
  2321  017600  000000                        0
  2322  017602  000000                        0
  2323  017604  110000                        LONGV                   ;BOX
  2324  017606  040000                        INTX
  2325  017610  021777                        MAXY
  2326  017612  041777                        INTX!MAXX
  2327  017614  000000                        0
  2328  017616  040000                        INTX
  2329  017620  021777                        MINUSX!MAXY
  2330  017622  061777                        INTX!MINUSX!MAXX
  2331  017624  000000                        0
  2332  017626  114000                        POINT
  2333  017630  000040                        40
  2334  017632  000000                        0
  2335  017634  110000                        LONGV
  2341  017636  060100                        INTX!MINUSX!100
   (1)  017640  000200                        200
   (1)  017642  040100                        INTX!100
   (1)  017644  000200                        200
   (1)  017646  060100                        INTX!MINUSX!100
   (1)  017650  000200                        200
   (1)  017652  040100                        INTX!100
   (1)  017654  000200                        200
   (1)  017656  060100                        INTX!MINUSX!100
   (1)  017660  000200                        200
   (1)  017662  040100                        INTX!100
   (1)  017664  000200                        200
   (1)  017666  060100                        INTX!MINUSX!100
   (1)  017670  000200                        200
   (1)  017672  040100                        INTX!100
   (1)  017674  000200                        200
```

```
  2342  017676  114000                  POINT
  2343  017700  000000                  0
  2344  017702  001737                  MAXY-40
  2345  017704  110000                  LONGV
  2351  017706  040200                  INTX!200
   (1)  017710  000100                  100
   (1)  017712  040200                  INTX!200
   (1)  017714  020100                  MINUSX!100
   (1)  017716  040200                  INTX!200
   (1)  017720  000100                  100
   (1)  017722  040200                  INTX!200
   (1)  017724  020100                  MINUSX!100
   (1)  017726  040200                  INTX!200
   (1)  017730  000100                  100
   (1)  017732  040200                  INTX!200
   (1)  017734  020100                  MINUSX!100
   (1)  017736  040200                  INTX!200
   (1)  017740  000100                  100
   (1)  017742  040200                  INTX!200
   (1)  017744  020100                  MINUSX!100
  2352  017746  114000                  POINT
  2353  017750  001737                  MAXX-40
  2354  017752  001777                  MAXY
  2355  017754  110000                  LONGV
  2361  017756  040100                  INTX!100
   (1)  017760  020200                  MINUSX!200
   (1)  017762  060100                  INTX!MINUSX!100
   (1)  017764  020200                  MINUSX!200
   (1)  017766  040100                  INTX!100
   (1)  017770  020200                  MINUSX!200
   (1)  017772  060100                  INTX!MINUSX!100
   (1)  017774  020200                  MINUSX!200
   (1)  017776  040100                  INTX!100
   (1)  020000  020200                  MINUSX!200
   (1)  020002  060100                  INTX!MINUSX!100
   (1)  020004  020200                  MINUSX!200
   (1)  020006  040100                  INTX!100
   (1)  020010  020200                  MINUSX!200
   (1)  020012  060100                  INTX!MINUSX!100
   (1)  020014  020200                  MINUSX!200
```

```
                                        POINT
                                        MAXX
                                        40
                                        LONGV
                                        INTX!MINUSX!200
                                        MINUSX!100
                                        INTX!MINUSX!200
                                        100
                                        INTX!MINUSX!200
                                        MINUSX!100
                                        INTX!MINUSX!200
                                        100
                                        INTX!MINUSX!200
                                        MINUSX!100
                                        INTX!MINUSX!200
                                        100
                                        INTX!MINUSX!200
                                        MINUSX!100
                                        INTX!MINUSX!200
                                        100
                                ;POSITION THE STARTING POINT OFF OF THE VIEWING SCREENE
                                        POINT
                                        MAXX/2
                                        0
                                        LONGV
                                        0
                                        MINUSX!200
                                ;NOW DRAW AN DIAMOND THAT INTERSECTS EACH OF THE FOUR EDGES
                                        LONGV
                                        INTX!1200
                                        1200
                                        INTX!MINUSX!1200
                                        1200
                                        INTX!MINUSX!1200
                                        MINUSX!1200
                                        INTX!1200
                                        MINUSX!1200

                                        .SBTTL  VECTOR SCALE SUB-PICTURE


                                        VCTR00!4
                                        POINT!INT7
                                        MAXX/2
                                        MAXY/2
                                PICSCA: VCTR00!4
                                        LONGV
                                        MINUSX!150
                                        MINUSY!150
                                        INTX!320
                                        0
                                        INTX
                                        320
                                        INTX!MINUSX!320
                                        0
```

```
2407   020160  040000                          INTX
2408   020162  020320                          MINUSX!320
2409   020164  154024                          VCTR00!4
2410   020166  173400                          DSTOP
2411   020170  160000                          DJMP
2412   020172  017572                          PICSCS
2413                                           .SBTTL  VECTOR STARTING SUB-PICTURE
2414
2415   020174  114000                  VSTRT:  POINT
2416   020176  001003                          1003
2417   020200  001200                          640.
2418   020202  110000                          LONGV                   ;VECTOR 1
2419   020204  040000                          INTX
2420   020206  000577                          383.
2421   020210  114000                          POINT
2422   020212  001003                          1003
2423   020214  001400                          768.
2424   020216  110000                          LONGV                   ;VECTOR 2
2425   020220  040200                          INTX!128.
2426   020222  000000                          0
2427   020224  114000                          POINT
2428   020226  001004                          1004
2429   020230  001366                          758.
2430   020232  110000                          LONGV                   ;VECTOR 3
2431   020234  040177                          INTX!127.
2432   020236  000000                          0
2433   020240  114000                          POINT
2434   020242  001003                          1003
2435   020244  001400                          768.
2436   020246  110000                          LONGV                   ;VECTOR 4
2437   020250  060200                          INTX!MINUSX!128.
2438   020252  000000                          0
2439   020254  114000                          POINT
2440   020256  001002                          1002
2441   020260  001366                          758.
2442   020262  110000                          LONGV                   ;VECTOR 5
2443   020264  060177                          INTX!MINUSX!127.
2444   020266  000000                          0
2445   020270  114000                          POINT
2446   020272  001003                          1003
2447   020274  001200                          640.
2448   020276  110000                          LONGV                   ;VECTOR 6
2449   020300  040200                          INTX!128.
2450   020302  000000                          0
2451   020304  114000                          POINT
2452   020306  001003                          1003
2453   020310  001200                          640.
2454   020312  110000                          LONGV                   ;VECTOR 7
2455   020314  040000                          INTX
2456   020316  020200                          MINUSX!128.
2457   020320  114000                          POINT
2458   020322  001003                          1003
2459   020324  001200                          640.
2460   020326  110000                          LONGV                   ;VECTOR 8
```

```
 2461  020330  060200                    INTX!MINUSX!128.
 2462  020332  000000                    0
 2463
 2464
 2465                              .SBTTL  MAJOR AXIS OFFSET SUB-PICTURE
 2466                              ;       +X +Y
 2467  020334  114000                      POINT
 2468  020336  001000                      1000
 2469  020340  000400                      400
 2470  020342  110000                      LONGV
 2471  020344  040177                      INTX!177                ;X MINOR
 2472  020346  000200                      200
 2473  020350  114000                      POINT
 2474  020352  001000                      1000
 2475  020354  000400                      400
 2476  020356  110000                      LONGV
 2477  020360  040200                      INTX!200                ;Y MINOR
 2478  020362  000177                      177
 2479                              ;       +X -Y
 2480
 2481  020364  114000                      POINT
 2482  020366  001000                      1000
 2483  020370  000400                      400
 2484  020372  110000                      LONGV
 2485  020374  040177                      INTX!177                ;X MINOR
 2486  020376  020200                      MINUSY!200
 2487  020400  114000                      POINT
 2488  020402  001000                      1000
 2489  020404  000400                      400
 2490  020406  110000                      LONGV
 2491  020410  040200                      INTX!200                ;Y MINOR
 2492  020412  020177                      MINUSX!177
 2493
 2494                              ;       -X -Y
 2495  020414  114000                      POINT
 2496  020416  001000                      1000
 2497  020420  000400                      400
 2498  020422  110000                      LONGV
 2499  020424  060177                      INTX!MINUSX!177         ;X MINOR
 2500  020426  020200                      MINUSY!200
 2501  020430  114000                      POINT
 2502  020432  001000                      1000
 2503  020434  000400                      400
 2504  020436  110000                      LONGV
 2505  020440  060200                      INTX!MINUSX!200         ;Y MINOR
 2506  020442  020177                      MINUSX!177
 2507
 2508                              ;       -X +Y
 2509  020444  114000                      POINT
 2510  020446  001000                      1000
 2511  020450  000400                      400
 2512  020452  110000                      LONGV
 2513  020454  060177                      INTX!MINUSX!177         ;X MINOR
 2514  020456  000200                      200
```

```
2515  020460  114000                          POINT
2516  020462  001000                          1000
2517  020464  000400                          400
2518  020466  110000                          LONGV
2519  020470  060200                          INTX!MINUSX!200
2520  020472  000177                          177
2521
2522  020474  173400                          DSTOP
2523  020476  160000                          DJMP
2524  020500  020174                          VSTRT
2525
2526
2527                                          .SBTTL  CHARACTER SCALE SUB-PICTURE
2528
2529                                  ;"A" CHARACTER
2530
2531  020502  164700                  CHAQU:  CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
2532  020504  114000                          POINT
2533  020506  000700                          700
2534  020510  001400                          1400
2535  020512  110000                          LONGV                           ;DRAW REF. LINE
2536  020514  040400                          INTX!400
2537  020516  000000                          0
2538  020520  114000                          POINT
2539  020522  000700                          700
2540  020524  001400                          1400
2541  020526  154340                          CHARS3                          ;CHAR SIZE 3 ( X2)
 (1)  020530  100000                          CHAR                            ;CHARACTER MODE
 (1)  020532     101                          .BYTE   101
 (1)  020533     000                          .BYTE   0
 (1)  020534  154300                          CHARS2                          ;CHAR SIZE 2 ( 1 1/2 X)
 (1)  020536  100000                          CHAR                            ;CHAR MODE
 (1)  020540     101                          .BYTE   101
 (1)  020541     000                          .BYTE   0
 (1)  020542  154240                          CHARS1                          ;CHAR SIZE 1 ( 1X)
 (1)  020544  100000                          CHAR
 (1)  020546     101                          .BYTE   101
 (1)  020547     000                          .BYTE   0
 (1)  020550  154200                          CHARS0                          ;CHAR SIZE ( 1/2)
 (1)  020552  100000                          CHAR
 (1)  020554     101                          .BYTE   101
 (1)  020555     000                          .BYTE   0
2542
2543                                  ;"B" CHARACTER
2544
2545  020556  114000                          POINT
2546  020560  000700                          700
2547  020562  001200                          1200
2548  020564  110000                          LONGV                           ;DRAW REF, LINE
2549  020566  040400                          INTX!400
2550  020570  000000                          0
2551  020572  114000                          POINT
2552  020574  000700                          700
2553  020576  001200                          1200
```

```
 2554  020600  154340                    CHARS3                  ;CHAR SIZE 3 ( X2)
  (1)  020602  100000                    CHAR                    ;CHARACTER MODE
  (1)  020604     102                    .BYTE   102
  (1)  020605     000                    .BYTE   0
  (1)  020606  154300                    CHARS2                  ;CHAR SIZE 2 ( 1 1/2 X)
  (1)  020610  100000                    CHAR                    ;CHAR MODE
  (1)  020612     102                    .BYTE   102
  (1)  020613     000                    .BYTE   0
  (1)  020614  154240                    CHARS1                  ;CHAR SIZE 1 ( 1X)
  (1)  020616  100000                    CHAR
  (1)  020620     102                    .BYTE   102
  (1)  020621     000                    .BYTE   0
  (1)  020622  154200                    CHARS0                  ;CHAR SIZE ( 1/2)
  (1)  020624  100000                    CHAR
  (1)  020626     102                    .BYTE   102
  (1)  020627     000                    .BYTE   0
 2555
 2556                           ;"F" CHARACTER
 2557
 2558  020630  114000                    POINT
 2559  020632  000700                    700
 2560  020634  001000                    1000
 2561  020636  110000                    LONGV                   ;DRAW REF. LINE
 2562  020640  040400                    INTX!400
 2563  020642  000000                    0
 2564  020644  114000                    POINT
 2565  020646  000700                    700
 2566  020650  001000                    1000
 2567  020652  154340                    CHARS3                  ;CHAR SIZE 3 ( X2)
  (1)  020654  100000                    CHAR                    ;CHARACTER MODE
  (1)  020656     106                    .BYTE   106
  (1)  020657     000                    .BYTE   0
  (1)  020660  154300                    CHARS2                  ;CHAR SIZE 2 ( 1 1/2 X)
  (1)  020662  100000                    CHAR                    ;CHAR MODE
  (1)  020664     106                    .BYTE   106
  (1)  020665     000                    .BYTE   0
  (1)  020666  154240                    CHARS1                  ;CHAR SIZE 1 ( 1X)
  (1)  020670  100000                    CHAR
  (1)  020672     106                    .BYTE   106
  (1)  020673     000                    .BYTE   0
  (1)  020674  154200                    CHARS0                  ;CHAR SIZE ( 1/2)
  (1)  020676  100000                    CHAR
  (1)  020700     106                    .BYTE   106
  (1)  020701     000                    .BYTE   0
 2568
 2569                           ;"O" CHARACTER
 2570
 2571  020702  117000                    POINT!INT4
 2572  020704  000700                    700
 2573  020706  000600                    600
 2574
 2575  020710  154340                    CHARS3                  ;CHAR SIZE 3 ( X2)
  (1)  020712  100000                    CHAR                    ;CHARACTER MODE
  (1)  020714     117                    .BYTE   117
```

```
   (1)  020715     000                .BYTE   0
   (1)  020716  154300     CHARS2                          ;CHAR SIZE 2 ( 1 1/2 X)
   (1)  020720  100000                CHAR                 ;CHAR MODE
   (1)  020722     117                .BYTE   117
   (1)  020723     000                .BYTE   0
   (1)  020724  154240     CHARS1                          ;CHAR SIZE 1 ( 1X)
   (1)  020726  100000                CHAR
   (1)  020730     117                .BYTE   117
   (1)  020731     000                .BYTE   0
   (1)  020732  154200     CHARS0                          ;CHAR SIZE ( 1/2)
   (1)  020734  100000                CHAR
   (1)  020736     117                .BYTE   117
   (1)  020737     000                .BYTE   0
  2576
  2577  020740  154024                VCTR00!4             ;LOAD VECTOR SCALE TO NORMAL SIZE
  2578  020742  114000                POINT
  2579  020744  000700                700
  2580  020746  000600                600
  2581  020750  154030                VCTR00!10            ;LOAD 2X VECTOR SIZE
  2582  020752  162000                DJSR                 ;DJSR TO DISPLAY SCALED POINTS AROUND THE "O"
  2583  020754  021154                ORELPT
  2584  020756  154026                VCTR00!6             ;LOAD VECTOR SCALE TO 1 1/2 SIZE
  2585  020760  162000                DJSR                 ;DJSR TO DISPLAY SCALED POINTS
  2586  020762  021154                ORELPT
  2587  020764  154024                VCTR00!4             ;LOAD VECTOR SCALE TO 1 SIZE
  2588  020766  162000                DJSR                 ;DJSR TO DISPLAY POINTS
  2589  020770  021154                ORELPT
  2590  020772  154022                VCTR00!2             ;LOAD VECTOR SCALE TO 1/2 SIZE
  2591  020774  162000                DJSR                 ;DJSR TO DISPLAY RELATIVE POINTTS
  2592  020776  021154                ORELPT
  2593  021000  154024                VCTR00!4             ;RETURN TO NORMAL SIZE
  2594  021002  164000                DNOP
  2595  021004  164000                DNOP
  2596  021006  164000                DNOP
  2597  021010  164000                DNOP
  2598  021012  164000                DNOP
  2599  021014  164000                DNOP
  2600  021016  164000                DNOP
  2601
  2602                              ;"T" CHARACTER
  2603
  2604  021020  114000                POINT
  2605  021022  000700                700
  2606  021024  000400                400
  2607  021026  110000                LONGV
  2608  021030  040400                INTX!400
  2609  021032  000000                0
  2610  021034  114000                POINT
  2611  021036  000700                700
  2612  021040  000400                400
  2613  021042  154340                CHARS3               ;CHAR SIZE 3 ( X2)
   (1)  021044  100000                CHAR                 ;CHARACTER MODE
   (1)  021046     124                .BYTE   124
   (1)  021047     000                .BYTE   0
```

```
   (1)  021050  154300                        CHARS2                          ;CHAR SIZE 2 ( 1 1/2 X)
   (1)  021052  100000                        CHAR                            ;CHAR MODE
   (1)  021054     124                        .BYTE   124
   (1)  021055     000                        .BYTE   0
   (1)  021056  154240                        CHARS1                          ;CHAR SIZE 1 ( 1X)
   (1)  021060  100000                        CHAR
   (1)  021062     124                        .BYTE   124
   (1)  021063     000                        .BYTE   0
   (1)  021064  154200                        CHARS0                          ;CHAR SIZE ( 1/2)
   (1)  021066  100000                        CHAR
   (1)  021070     124                        .BYTE   124
   (1)  021071     000                        .BYTE   0
  2614                              ;"X" CHARACTER
  2615  021072  114000                        POINT
  2616  021074  000700                        700
  2617  021076  000200                        200
  2618  021100  110000                        LONGV
  2619  021102  040400                        INTX!400
  2620  021104  000000                        0
  2621  021106  114000                        POINT
  2622  021110  000700                        700
  2623  021112  000200                        200
  2624  021114  154340                        CHARS3                          ;CHAR SIZE 3 ( X2)
   (1)  021116  100000                        CHAR                            ;CHARACTER MODE
   (1)  021120     130                        .BYTE   130
   (1)  021121     000                        .BYTE   0
   (1)  021122  154300                        CHARS2                          ;CHAR SIZE 2 ( 1 1/2 X)
   (1)  021124  100000                        CHAR                            ;CHAR MODE
   (1)  021126     130                        .BYTE   130
   (1)  021127     000                        .BYTE   0
   (1)  021130  154240                        CHARS1                          ;CHAR SIZE 1 ( 1X)
   (1)  021132  100000                        CHAR
   (1)  021134     130                        .BYTE   130
   (1)  021135     000                        .BYTE   0
   (1)  021136  154200                        CHARS0                          ;CHAR SIZE ( 1/2)
   (1)  021140  100000                        CHAR
   (1)  021142     130                        .BYTE   130
   (1)  021143     000                        .BYTE   0
  2625  021144  154240                        CHARS1
  2626  021146  173400                        DSTOP
  2627  021150  160000                        DJMP
  2628  021152  020502                        CHAQU
  2629
  2630  021154  130000              ORELPT:   RELATP                          ;ENABLE RELATIVE POINT MODE
  2631  021156  041600                        INTX!1600
  2632  021160  040013                        INTX!13
  2633  021162  061600                        INTX!MINUSX!1600
  2634  021164  040113                        INTX!113
  2635  021166  003400                        3400
  2636  021170  166000                        DPOP
  2637
  2638
  2639                              .SBTTL  ROTATE CHARACTERS SUBPICTURE
  2640
```

```
 2642  021172  170003                  ROTCHR: DMENU1                  ;ENABLE MENU
 2643  021174  114000                          POINT
 2644  021176  000000                          0
 2645  021200  000000                          0
 2646  021202  120000                          BASICV                  ;DRAW REF. BOX
 2647  021204  042177                          INTX!PATH0!177
 2648  021206  053777                          INTX!PATH2!MAXY
 2649  021210  062177                          INTX!PATH4!177
 2650  021212  073777                          INTX!PATH6!MAXY
 2651  021214  114000                          POINT
 2652  021216  000050                          50
 2653  021220  000000                          0
 2654  021222  155400                          CHRRT1                  ;ENABLE CHAR ROTATION
 2661  021224  163005                          DJSRR!WHERE2            ;DJSR RELATIVE TO THE TAG "CHARQA"
 2662  021226  155000                          CHRRT0                  ;DISABLE ROTATION
 2663  021230  170002                          DMENU0                  ;RETURN TO MAIN SCREEN
 2664  021232  173400                          DSTOP
 2665  021234  160000                          DJMP                    ;JUMP BACK TO MAIN TEXT
 2666  021236  026070                          BUFFER
 2667
 2668                                  ;TWO COPIES OF THE "QUICK BROWN FOX" MESSAGE
 2669
 2670  021240  170040                  CHARQA: STATSA!ITAL0    ;NON ITALIC
 2671  021242  100000                          CHAR
 2677  021244  044124  020105  052521          .ASCII  /THE QUICK BROWN FOX JUMPS OVER THE LAZY DOGS /
  (2)  021252  041511  020113  051102
  (2)  021260  053517  020116  047506
  (2)  021266  020130  052512  050115
  (2)  021274  020123  053117  051105
  (2)  021302  052040  042510  046040
  (2)  021310  055101  020131  047504
  (2)  021316  051507     040
  (2)  021321     124  042510  050440          .ASCII  /THE QUICK BROWN FOX JUMPS OVER THE LAZY DOGS /
  (2)  021326  044525  045503  041040
  (2)  021334  047522  047127  043040
  (2)  021342  054117  045040  046525
  (2)  021350  051520  047440  042526
  (2)  021356  020122  044124  020105
  (2)  021364  040514  054532  042040
  (2)  021372  043517  020123
 2678  021376     015     012                  .BYTE   15,12
 2679  021400  170060                          STATSA!ITAL1
 2680  021402  044124  020105  052521          .ASCII  /THE QUICK BROWN FOX JUMPS OVER THE LAZY DOGS /
  (2)  021410  041511  020113  051102
  (2)  021416  053517  020116  047506
  (2)  021424  020130  052512  050115
  (2)  021432  020123  053117  051105
  (2)  021440  052040  042510  046040
  (2)  021446  055101  020131  047504
  (2)  021454  051507     040
  (2)  021457     124  042510  050440          .ASCII  /THE QUICK BROWN FOX JUMPS OVER THE LAZY DOGS /
  (2)  021464  044525  045503  041040
  (2)  021472  047522  047127  043040
  (2)  021500  054117  045040  046525
```

```
   (2)  021506  051520  047440  042526
   (2)  021514  020122  044124  020105
   (2)  021522  040514  054532  042040
   (2)  021530  043517  020123
2681    021534     015     012                .BYTE   15,12
2682
2683                                    ;LOWER CASE ASCII MESSAGES
2684
2685    021536  170040                  CHARQD: STATSA!ITALO
2686    021540  100000                          CHAR
2695    021542     164     150     145          .BYTE   164,150,145,40,161,165,151,143,153,40,142,162,157,167,156
   (2)  021545     040     161     165
   (2)  021550     151     143     153
   (2)  021553     040     142     162
   (2)  021556     157     167     156
   (2)  021561     040     146     157          .BYTE   40,146,157,170,40,152,165,155,160,163,40,157,166,145,162
   (2)  021564     170     040     152
   (2)  021567     165     155     160
   (2)  021572     163     040     157
   (2)  021575     166     145     162
   (2)  021600     040     164     150          .BYTE   40,164,150,145,40,154,141,172,171,40
   (2)  021603     145     040     154
   (2)  021606     141     172     171
   (2)  021611     040
   (2)  021612     144     157     147          .BYTE   144,157,147,163,40
   (2)  021615     163     040
   (2)  021617     164     150     145          .BYTE   164,150,145,40,161,165,151,143,153,40,142,162,157,167,156
   (2)  021622     040     161     165
   (2)  021625     151     143     153
   (2)  021630     040     142     162
   (2)  021633     157     167     156
   (2)  021636     040     146     157          .BYTE   40,146,157,170,40,152,165,155,160,163,40,157,166,145,162
   (2)  021641     170     040     152
   (2)  021644     165     155     160
   (2)  021647     163     040     157
   (2)  021652     166     145     162
   (2)  021655     040     164     150          .BYTE   40,164,150,145,40,154,141,172,171,40
   (2)  021660     145     040     154
   (2)  021663     141     172     171
   (2)  021666     040
   (2)  021667     144     157     147          .BYTE   144,157,147,163,40
   (2)  021672     163     040
2696    021674     015     012                .BYTE   15,12
2697    021676  170060                  STATSA!ITALI                    ;SET ITALICS
2698    021700     164     150     145          .BYTE   164,150,145,40,161,165,151,143,153,40,142,162,157,167,156
   (2)  021703     040     161     165
   (2)  021706     151     143     153
   (2)  021711     040     142     162
   (2)  021714     157     167     156
   (2)  021717     040     146     157          .BYTE   40,146,157,170,40,152,165,155,160,163,40,157,166,145,162
   (2)  021722     170     040     152
   (2)  021725     165     155     160
   (2)  021730     163     040     157
   (2)  021733     166     145     162
```

```
    (2)  021736      040      164      150          .BYTE   40,164,150,145,40,154,141,172,171,40
    (2)  021741      145      040      154
    (2)  021744      141      172      171
    (2)  021747      040
    (2)  021750      144      157      147          .BYTE   144,157,147,163,40
    (2)  021753      163      040
    (2)  021755      164      150      145          .BYTE   164,150,145,40,161,165,151,143,153,40,142,162,157,167,156
    (2)  021760      040      161      165
    (2)  021763      151      143      153
    (2)  021766      040      142      162
    (2)  021771      157      167      156
    (2)  021774      040      146      157          .BYTE   40,146,157,170,40,152,165,155,160,163,40,157,166,145,162
    (2)  021777      170      040      152
    (2)  022002      165      155      160
    (2)  022005      163      040      157
    (2)  022010      166      145      162
    (2)  022013      040      164      150          .BYTE   40,164,150,145,40,154,141,172,171,40
    (2)  022016      145      040      154
    (2)  022021      141      172      171
    (2)  022024      040
    (2)  022025      144      157      147          .BYTE   144,157,147,163,40
    (2)  022030      163      040
 2699    022032      015      012                   .BYTE   15,12
 2700    022034   170040                            STATSA!ITALO
 2701    022036   166000                            DPOP
 2702
 2703                                               .SBTTL
 2704                                               .SBTTL  LIGHT-PEN SUBPICTURE
 2705                                               .SBTTL
 2706
 2707
 2708                                               .SBTTL  POSITION THE OCTAGON
 2709    022040   164774                    FRME16: CONSL1!BIT7!BIT6!BIT5!BIT4!BIT3!BIT2    ;ENABLE CONSOLE #1
 2710    022042   164374                            CONSL0!BIT7!BIT6!BIT5!BIT4!BIT3!BIT2
 2711    022044   114140                            POINT!LPON
 2712    022046   001400                            1400
 2713    022050   001200                            1200
 2714    022052   110000                            LONGV                           ;OCTOGON BY LENGTH OF 137
  (1)    022054   040137                            INTX+137
  (1)    022056   000000                            0
  (1)    022060   040137                            INTX+137
  (1)    022062   000137                            137
  (1)    022064   040000                            INTX
  (1)    022066   000137                            137
  (1)    022070   060137                            INTX!MINUSX+137
  (1)    022072   000137                            137
  (1)    022074   060137                            INTX!MINUSX+137
  (1)    022076   000000                            0
  (1)    022100   060137                            INTX!MINUSX+137
  (1)    022102   020137                            MINUSX+137
  (1)    022104   040000                            INTX
  (1)    022106   020137                            MINUSX+137
  (1)    022110   040137                            INTX+137
  (1)    022112   020137                            MINUSX+137
```

```
                                        .SBTTL  DISPLAY ON CONSOLE #0 THE X-Y READOUT VALUE
022114  164640                                  CONSL1!BIT7!BIT5                ;DISABLE CONSOLE #1
022116  114000                                  POINT
022120  001300                                  1300
022122  001500                                  1500
022124  100000                                  CHAR
022126  036530                                  .ASCII  /X=/
022130  030061  030060          DLT14A:         .ASCII  /1000/
022134     040     040     040                  .BYTE   40,40,40
022137     131  020075                         .ASCII  /Y= /
022142  030061  030060          DLT14B:         .ASCII  /1000/
022146  114000                                  POINT
022150  001250                                  1250
022152  001340                                  1340
022154  100000                                  CHAR
022156  160000                                  DJMP
022160  022270                  MSOPEN:         PENOF0          ;JUMP TO PEN SWITCH MESSAGE FOR CONSOLE #0
                                        .SBTTL  DISPLAY ON CONSOLE #1 THE X-Y READOUT VALUE
022162  164760                  LPRTA:          CONSL1!BIT7!BIT6!BIT5!BIT4      ;ENABLE CONSOLE #1
022164  164240                                  CONSL0!BIT7!BIT5                ;DISABLE CONSOLE #0
022166  114000                                  POINT
022170  001300                                  1300
022172  001500                                  1500                            ;POSITION THE X-Y MESSAGE
022174  100000                                  CHAR
022176  036530                                  .ASCII  /X=/
022200  030061  030060          DLT14C:         .ASCII  /1000/
022204     040     040     040                  .BYTE   40,40,40
022207     131  020075                         .ASCII  /Y= /
022212  030061  030060          DLT14D:         .ASCII  /1000/
022216  114000                                  POINT
022220  001250                                  1250
022222  001340                                  1340                            ;POSITION THE PEN SWITCH MESSAGE FOR CONSOLE #1
022224  100000                                  CHAR
022226  160000                                  DJMP
022230  022350                  MS1PEN:         PENOF1                          ;JUMP TO MESSAGE FOR #1
                                        .SBTTL  DISPLAY HIT COUNT MESSAGE
022232  117140                  LPRTC:          POINT!INT4!LPON
022234  001300                                  1300
022236  000200                                  200
022240  164360                                  CONSL0!BIT7!BIT6!BIT5!BIT4      ;ENABLE CONSOLE #0
022242  100000                                  CHAR
022244  044510  020124  047503                  .ASCII  /HIT COUNT = 0000/
022252  047125  020124  020075
022260  030060  030060
022264  160000                  FRM16B:         DJMP
022266  022430                                  FRM16C

022270  042520  020116  053523  PENOF0:         .ASCII  /PEN SWITCH #0 IS OFF/
022276  052111  044103  021440
022304  020060  051511  047440
022312  043106
```

```
     022314  160000                                 DJMP
     022316  022162                                 LPRTA
     022320  042520  020116  053523  PENON0:  .ASCII  /PEN SWITCH #0 IS ON /
     022326  052111  044103  021440
     022334  020060  051511  047440
     022342  020116
     022344  160000                                 DJMP
     022346  022162                                 LPRTA
     022350  042520  020116  053523  PENOF1:  .ASCII  /PEN SWITCH #1 IS OFF/
     022356  052111  044103  021440
     022364  020061  051511  047440
     022372  043106
     022374  160000                                 DJMP
     022376  022232                                 LPRTC
     022400  042520  020116  053523  PENON1:  .ASCII  /PEN SWITCH #1 IS ON /
     022406  052111  044103  021440
     022414  020061  051511  047440
     022422  020116
     022424  160000                                 DJMP
     022426  022232                                 LPRTC

                                                    .SBTTL  HORIZONTAL REF. LINE SECTION

     022430  114000                          FRM16C: POINT
     022432  000000                                  0
     022434  000700                                  700
     022436  110000                                  LONGV
     022440  041777                                  INTX!MAXX
     022442  000000                                  0

     022444  114000                                  POINT                  ;POINT TO  X CORDINATE "0"
 (1) 022446  000000                                  0
 (1) 022450  000640                                  640                    ;Y CORD. = 640
 (1) 022452  110000                                  LONGV                  ;DRAW 30 UNIT VERTICAL LINE
 (1) 022454  040000                                  INTX
 (1) 022456  000030                                  30
     022460  114000                                  POINT                  ;POINT TO  X CORDINATE "200"
 (1) 022462  000200                                  200
 (1) 022464  000640                                  640                    ;Y CORD. = 640
 (1) 022466  110000                                  LONGV                  ;DRAW 30 UNIT VERTICAL LINE
 (1) 022470  040000                                  INTX
 (1) 022472  000030                                  30
     022474  114000                                  POINT                  ;POINT TO  X CORDINATE "400"
 (1) 022476  000400                                  400
 (1) 022500  000640                                  640                    ;Y CORD. = 640
 (1) 022502  110000                                  LONGV                  ;DRAW 30 UNIT VERTICAL LINE
 (1) 022504  040000                                  INTX
 (1) 022506  000030                                  30
     022510  114000                                  POINT                  ;POINT TO  X CORDINATE "600"
 (1) 022512  000600                                  600
 (1) 022514  000640                                  640                    ;Y CORD. = 640
 (1) 022516  110000                                  LONGV                  ;DRAW 30 UNIT VERTICAL LINE
 (1) 022520  040000                                  INTX
 (1) 022522  000030                                  30
```

```
 2797  022524  114000                POINT                   ;POINT TO  X CORDINATE "1000"
  (1)  022526  001000                1000
  (1)  022530  000640                640                     ;Y CORD. = 640
  (1)  022532  110000                LONGV                   ;DRAW 30 UNIT VERTICAL LINE
  (1)  022534  040000                INTX
  (1)  022536  000030                30
 2798  022540  114000                POINT                   ;POINT TO  X CORDINATE "1200"
  (1)  022542  001200                1200
  (1)  022544  000640                640                     ;Y CORD. = 640
  (1)  022546  110000                LONGV                   ;DRAW 30 UNIT VERTICAL LINE
  (1)  022550  040000                INTX
  (1)  022552  000030                30
 2799  022554  114000                POINT                   ;POINT TO  X CORDINATE "1400"
  (1)  022556  001400                1400
  (1)  022560  000640                640                     ;Y CORD. = 640
  (1)  022562  110000                LONGV                   ;DRAW 30 UNIT VERTICAL LINE
  (1)  022564  040000                INTX
  (1)  022566  000030                30
 2800  022570  114000                POINT                   ;POINT TO  X CORDINATE "1600"
  (1)  022572  001600                1600
  (1)  022574  000640                640                     ;Y CORD. = 640
  (1)  022576  110000                LONGV                   ;DRAW 30 UNIT VERTICAL LINE
  (1)  022600  040000                INTX
  (1)  022602  000030                30
 2801  022604  114000                POINT                   ;POINT TO  X CORDINATE "1777"
  (1)  022606  001777                1777
  (1)  022610  000640                640                     ;Y CORD. = 640
  (1)  022612  110000                LONGV                   ;DRAW 30 UNIT VERTICAL LINE
  (1)  022614  040000                INTX
  (1)  022616  000030                30
 2802
 2822                                .SBTTL  VERTICAL SPACEING SECTION
 2823
 2824  022620  114000                POINT
 2825  022622  000200                200
 2826  022624  000010                10
 2827  022626  100000                CHAR
 2828  022630  020130  047503  051117  .ASCII  /X COORD = 200 /
       022636  020104  020075  030062
       022644  020060
 2829  022646  114000                POINT
 2830  022650  000200                200
 2831  022652  000060                60
 2832
 2833  022654  110000                LONGV                   ;DRAW LOWER LINE
  (1)  022656  040200                INTX!200
  (1)  022660  000000                0
  (1)  022662  000000                0
  (1)  022664  000011                9.
  (1)  022666  060200                INTX!MINUSX!200         ;DRAW NEXT HIGHER LINE
  (1)  022670  000000                0
  (1)  022672  000000                0
  (1)  022674  000011                9.
  (1)  022676  040200                INTX!200                ;DRAW NEXT HIGHER LINE
```

```
   (1)  022700  000000          0
   (1)  022702  000000          0
   (1)  022704  000011          9.
   (1)  022706  060200          INTX!MINUSX!200                ;DRAW UPPER LINE
   (1)  022710  000000          0
   (1)  022712  000000          0
   (1)  022714  000040          40                             ;OFFSET FOR NEXT LINE
2834    022716  110000          LONGV                          ;DRAW LOWER LINE
   (1)  022720  040200          INTX!200
   (1)  022722  000000          0
   (1)  022724  000000          0
   (1)  022726  000010          8.
   (1)  022730  060200          INTX!MINUSX!200                ;DRAW NEXT HIGHER LINE
   (1)  022732  000000          0
   (1)  022734  000000          0
   (1)  022736  000010          8.
   (1)  022740  040200          INTX!200                       ;DRAW NEXT HIGHER LINE
   (1)  022742  000000          0
   (1)  022744  000000          0
   (1)  022746  000010          8.
   (1)  022750  060200          INTX!MINUSX!200                ;DRAW UPPER LINE
   (1)  022752  000000          0
   (1)  022754  000000          0
   (1)  022756  000040          40                             ;OFFSET FOR NEXT LINE
2835    022760  110000          LONGV                          ;DRAW LOWER LINE
   (1)  022762  040200          INTX!200
   (1)  022764  000000          0
   (1)  022766  000000          0
   (1)  022770  000007          7
   (1)  022772  060200          INTX!MINUSX!200                ;DRAW NEXT HIGHER LINE
   (1)  022774  000000          0
   (1)  022776  000000          0
   (1)  023000  000007          7
   (1)  023002  040200          INTX!200                       ;DRAW NEXT HIGHER LINE
   (1)  023004  000000          0
   (1)  023006  000000          0
   (1)  023010  000007          7
   (1)  023012  060200          INTX!MINUSX!200                ;DRAW UPPER LINE
   (1)  023014  000000          0
   (1)  023016  000000          0
   (1)  023020  000040          40                             ;OFFSET FOR NEXT LINE
2836    023022  110000          LONGV                          ;DRAW LOWER LINE
   (1)  023024  040200          INTX!200
   (1)  023026  000000          0
   (1)  023030  000000          0
   (1)  023032  000006          6
   (1)  023034  060200          INTX!MINUSX!200                ;DRAW NEXT HIGHER LINE
   (1)  023036  000000          0
   (1)  023040  000000          0
   (1)  023042  000006          6
   (1)  023044  040200          INTX!200                       ;DRAW NEXT HIGHER LINE
   (1)  023046  000000          0
   (1)  023050  000000          0
   (1)  023052  000006          6
```

```
   (1)  023054  060200  INTX!MINUSX!200                ;DRAW UPPER LINE
   (1)  023056  000000  0
   (1)  023060  000000  0
   (1)  023062  000040  40                             ;OFFSET FOR NEXT LINE
2837    023064  110000  LONGV                           ;DRAW LOWER LINE
   (1)  023066  040200  INTX!200
   (1)  023070  000000  0
   (1)  023072  000000  0
   (1)  023074  000005  5
   (1)  023076  060200  INTX!MINUSX!200                ;DRAW NEXT HIGHER LINE
   (1)  023100  000000  0
   (1)  023102  000000  0
   (1)  023104  000005  5
   (1)  023106  040200  INTX!200                       ;DRAW NEXT HIGHER LINE
   (1)  023110  000000  0
   (1)  023112  000000  0
   (1)  023114  000005  5
   (1)  023116  060200  INTX!MINUSX!200                ;DRAW UPPER LINE
   (1)  023120  000000  0
   (1)  023122  000000  0
   (1)  023124  000040  40                             ;OFFSET FOR NEXT LINE
2838    023126  110000  LONGV                           ;DRAW LOWER LINE
   (1)  023130  040200  INTX!200
   (1)  023132  000000  0
   (1)  023134  000000  0
   (1)  023136  000004  4
   (1)  023140  060200  INTX!MINUSX!200                ;DRAW NEXT HIGHER LINE
   (1)  023142  000000  0
   (1)  023144  000000  0
   (1)  023146  000004  4
   (1)  023150  040200  INTX!200                       ;DRAW NEXT HIGHER LINE
   (1)  023152  000000  0
   (1)  023154  000000  0
   (1)  023156  000004  4
   (1)  023160  060200  INTX!MINUSX!200                ;DRAW UPPER LINE
   (1)  023162  000000  0
   (1)  023164  000000  0
   (1)  023166  000040  40                             ;OFFSET FOR NEXT LINE
2839    023170  110000  LONGV                           ;DRAW LOWER LINE
   (1)  023172  040200  INTX!200
   (1)  023174  000000  0
   (1)  023176  000000  0
   (1)  023200  000003  3
   (1)  023202  060200  INTX!MINUSX!200                ;DRAW NEXT HIGHER LINE
   (1)  023204  000000  0
   (1)  023206  000000  0
   (1)  023210  000003  3
   (1)  023212  040200  INTX!200                       ;DRAW NEXT HIGHER LINE
   (1)  023214  000000  0
   (1)  023216  000000  0
   (1)  023220  000003  3
   (1)  023222  060200  INTX!MINUSX!200                ;DRAW UPPER LINE
   (1)  023224  000000  0
   (1)  023226  000000  0
```

```
   (1)  023230  000040                    40                              ;OFFSET FOR NEXT LINE
  2840  023232  110000                    LONGV                           ;DRAW LOWER LINE
   (1)  023234  040200                    INTX!200
   (1)  023236  000000                    0
   (1)  023240  000000                    0
   (1)  023242  000002                    2
   (1)  023244  060200                    INTX!MINUSX!200                 ;DRAW NEXT HIGHER LINE
   (1)  023246  000000                    0
   (1)  023250  000000                    0
   (1)  023252  000002                    2
   (1)  023254  040200                    INTX!200                        ;DRAW NEXT HIGHER LINE
   (1)  023256  000000                    0
   (1)  023260  000000                    0
   (1)  023262  000002                    2
   (1)  023264  060200                    INTX!MINUSX!200                 ;DRAW UPPER LINE
   (1)  023266  000000                    0
   (1)  023270  000000                    0
   (1)  023272  000040                    40                              ;OFFSET FOR NEXT LINE
  2841                                    .SBTTL  VARIABLE HORIZ. LINE LENGTH
  2842
  2843  023274  114000                    POINT
  2844  023276  001000                    1000
  2845  023300  000020                    20
  2846  023302  100000                    CHAR
  2847  023304  020130  047503  051117    .ASCII  /X COORDINATE = 1000 /
        023312  044504  040516  042524
        023320  036440  030440  030060
        023326  020060
  2848          000001                    L=1
  2849          000600                    M=600
  2861  023330  114000                    POINT                           ;POINT TO Y CORD. " M "
   (1)  023332  001000                    1000
   (1)  023334  000600                    M
   (1)  023336  110000                    LONGV                           ;DRAW A VECTOR " L " UNITS LONG
   (1)  023340  040001                    INTX! L
   (1)  023342  000000                    0
   (1)  023344  114000                    POINT                           ;POINT TO Y CORD. " M "
   (1)  023346  001000                    1000
   (1)  023350  000560                    M
   (1)  023352  110000                    LONGV                           ;DRAW A VECTOR " L " UNITS LONG
   (1)  023354  040002                    INTX! L
   (1)  023356  000000                    0
   (1)  023360  114000                    POINT                           ;POINT TO Y CORD. " M "
   (1)  023362  001000                    1000
   (1)  023364  000540                    M
   (1)  023366  110000                    LONGV                           ;DRAW A VECTOR " L " UNITS LONG
   (1)  023370  040003                    INTX! L
   (1)  023372  000000                    0
   (1)  023374  114000                    POINT                           ;POINT TO Y CORD. " M "
   (1)  023376  001000                    1000
   (1)  023400  000520                    M
   (1)  023402  110000                    LONGV                           ;DRAW A VECTOR " L " UNITS LONG
   (1)  023404  040004                    INTX! L
   (1)  023406  000000                    0
```

```
(1)  023410  114000  POINT                   ;POINT TO Y CORD. " M "
(1)  023412  001000  1000
(1)  023414  000500  M
(1)  023416  110000  LONGV                   ;DRAW A VECTOR " L " UNITS LONG
(1)  023420  040005  INTX! L
(1)  023422  000000  0
(1)  023424  114000  POINT                   ;POINT TO Y CORD. " M "
(1)  023426  001000  1000
(1)  023430  000460  M
(1)  023432  110000  LONGV                   ;DRAW A VECTOR " L " UNITS LONG
(1)  023434  040006  INTX! L
(1)  023436  000000  0
(1)  023440  114000  POINT                   ;POINT TO Y CORD. " M "
(1)  023442  001000  1000
(1)  023444  000440  M
(1)  023446  110000  LONGV                   ;DRAW A VECTOR " L " UNITS LONG
(1)  023450  040007  INTX! L
(1)  023452  000000  0
(1)  023454  114000  POINT                   ;POINT TO Y CORD. " M "
(1)  023456  001000  1000
(1)  023460  000420  M
(1)  023462  110000  LONGV                   ;DRAW A VECTOR " L " UNITS LONG
(1)  023464  040010  INTX! L
(1)  023466  000000  0
(1)  023470  114000  POINT                   ;POINT TO Y CORD. " M "
(1)  023472  001000  1000
(1)  023474  000400  M
(1)  023476  110000  LONGV                   ;DRAW A VECTOR " L " UNITS LONG
(1)  023500  040011  INTX! L
(1)  023502  000000  0
(1)  023504  114000  POINT                   ;POINT TO Y CORD. " M "
(1)  023506  001000  1000
(1)  023510  000360  M
(1)  023512  110000  LONGV                   ;DRAW A VECTOR " L " UNITS LONG
(1)  023514  040012  INTX! L
(1)  023516  000000  0
(1)  023520  114000  POINT                   ;POINT TO Y CORD. " M "
(1)  023522  001000  1000
(1)  023524  000340  M
(1)  023526  110000  LONGV                   ;DRAW A VECTOR " L " UNITS LONG
(1)  023530  040013  INTX! L
(1)  023532  000000  0
(1)  023534  114000  POINT                   ;POINT TO Y CORD. " M "
(1)  023536  001000  1000
(1)  023540  000320  M
(1)  023542  110000  LONGV                   ;DRAW A VECTOR " L " UNITS LONG
(1)  023544  040014  INTX! L
(1)  023546  000000  0
(1)  023550  114000  POINT                   ;POINT TO Y CORD. " M "
(1)  023552  001000  1000
(1)  023554  000300  M
(1)  023556  110000  LONGV                   ;DRAW A VECTOR " L " UNITS LONG
(1)  023560  040015  INTX! L
(1)  023562  000000  0
```

```
   (1)  023564  114000                    POINT                           ;POINT TO Y CORD. " M "
   (1)  023566  001000                    1000
   (1)  023570  000260                    M
   (1)  023572  110000                    LONGV                           ;DRAW A VECTOR " L " UNITS LONG
   (1)  023574  040016                    INTX! L
   (1)  023576  000000                    0
   (1)  023600  114000                    POINT                           ;POINT TO Y CORD. " M "
   (1)  023602  001000                    1000
   (1)  023604  000240                    M
   (1)  023606  110000                    LONGV                           ;DRAW A VECTOR " L " UNITS LONG
   (1)  023610  040017                    INTX! L
   (1)  023612  000000                    0
   (1)  023614  114000                    POINT                           ;POINT TO Y CORD. " M "
   (1)  023616  001000                    1000
   (1)  023620  000220                    M
   (1)  023622  110000                    LONGV                           ;DRAW A VECTOR " L " UNITS LONG
   (1)  023624  040020                    INTX! L
   (1)  023626  000000                    0
   (1)  023630  114000                    POINT                           ;POINT TO Y CORD. " M "
   (1)  023632  001000                    1000
   (1)  023634  000200                    M
   (1)  023636  110000                    LONGV                           ;DRAW A VECTOR " L " UNITS LONG
   (1)  023640  040021                    INTX! L
   (1)  023642  000000                    0
   (1)  023644  114000                    POINT                           ;POINT TO Y CORD. " M "
   (1)  023646  001000                    1000
   (1)  023650  000160                    M
   (1)  023652  110000                    LONGV                           ;DRAW A VECTOR " L " UNITS LONG
   (1)  023654  040022                    INTX! L
   (1)  023656  000000                    0
   (1)  023660  114000                    POINT                           ;POINT TO Y CORD. " M "
   (1)  023662  001000                    1000
   (1)  023664  000140                    M
   (1)  023666  110000                    LONGV                           ;DRAW A VECTOR " L " UNITS LONG
   (1)  023670  040023                    INTX! L
   (1)  023672  000000                    0
   (1)  023674  114000                    POINT                           ;POINT TO Y CORD. " M "
   (1)  023676  001000                    1000
   (1)  023700  000120                    M
   (1)  023702  110000                    LONGV                           ;DRAW A VECTOR " L " UNITS LONG
   (1)  023704  040024                    INTX! L
   (1)  023706  000000                    0
2862                                      .SBTTL  INTENSITY LEVEL SECTION OF LIGHT PEN TEST
2863
2864  023710  114000                      POINT
2865  023712  000200                      200
2866  023714  001740                      1740
2867  023716  100000                      CHAR                            ;CHAR MODE
2868  023720  036530  031040  030060      .ASCII  /X= 200/
2869  023726  114000                      POINT
2870  023730  000700                      700
2871  023732  001740                      1740
2872  023734  100000                      CHAR
2873  023736  036530  030061  030060      .ASCII  /X=1000/
```

```
 2874  023744  114000                          POINT
 2875  023746  001100                          1100
 2876  023750  001740                          1740
 2877  023752  100000                          CHAR
 2878  023754  036530  030061  033467          .ASCII  /X=1077/
 2879
 2880          001000                          J=1000
 2881  023762  114000                          POINT
 2882  023764  000020                          20
 2883  023766  001700                          1700
 2884  023770  103600                          CHAR!INT7                       ;CHAR MODE
 2885  023772  036531  033461  030060          .ASCII  /Y=1700/
 2886  024000  114000                          POINT
 2887  024002  000200                          200
 2888  024004  001700                          1700
 2889  024006  110000                          LONGV                   ;DRAW HORIZ. LINE
 2890  024010  040600                          INTX!600
 2891  024012  000000                          0
 2892  024014  130000                          RELATP
 2893  024016  057600                          57600
 2894  024020  114000                          POINT
 2895  024022  000020                          20
 2896  024024  001600                          1600
 2897  024026  103400                          CHAR!INT6                       ;CHAR MODE
 2898  024030  036531  033061  030060          .ASCII  /Y=1600/
 2899  024036  114000                          POINT
 2900  024040  000200                          200
 2901  024042  001600                          1600
 2902  024044  110000                          LONGV                   ;DRAW HORIZ. LINE
 2903  024046  040600                          INTX!600
 2904  024050  000000                          0
 2905  024052  130000                          RELATP
 2906  024054  057600                          57600
 2907  024056  114000                          POINT
 2908  024060  000020                          20
 2909  024062  001500                          1500
 2910  024064  103200                          CHAR!INT5                       ;CHAR MODE
 2911  024066  036531  032461  030060          .ASCII  /Y=1500/
 2912  024074  114000                          POINT
 2913  024076  000200                          200
 2914  024100  001500                          1500
 2915  024102  110000                          LONGV                   ;DRAW HORIZ. LINE
 2916  024104  040600                          INTX!600
 2917  024106  000000                          0
 2918  024110  130000                          RELATP
 2919  024112  057600                          57600
 2920  024114  114000                          POINT
 2921  024116  000020                          20
 2922  024120  001400                          1400
 2923  024122  103000                          CHAR!INT4                       ;CHAR MODE
 2924  024124  036531  032061  030060          .ASCII  /Y=1400/
 2925  024132  114000                          POINT
 2926  024134  000200                          200
 2927  024136  001400                          1400
```

```
 2928   024140  110000                          LONGV                   ;DRAW HORIZ. LINE
 2929   024142  040600                          INTX!600
 2930   024144  000000                          0
 2931   024146  130000                          RELATP
 2932   024150  057600                          57600
 2933   024152  114000                          POINT
 2934   024154  000020                          20
 2935   024156  001300                          1300
 2936   024160  102600                          CHAR!INT3                       ;CHAR MODE
 2937   024162  036531  031461  030060          .ASCII  /Y=1300/
 2938   024170  114000                          POINT
 2939   024172  000200                          200
 2940   024174  001300                          1300
 2941   024176  110000                          LONGV                   ;DRAW HORIZ. LINE
 2942   024200  040600                          INTX!600
 2943   024202  000000                          0
 2944   024204  130000                          RELATP
 2945   024206  057600                          57600
 2946   024210  114000                          POINT
 2947   024212  000020                          20
 2948   024214  001200                          1200
 2949   024216  102400                          CHAR!INT2                       ;CHAR MODE
 2950   024220  036531  031061  030060          .ASCII  /Y=1200/
 2951   024226  114000                          POINT
 2952   024230  000200                          200
 2953   024232  001200                          1200
 2954   024234  110000                          LONGV                   ;DRAW HORIZ. LINE
 2955   024236  040600                          INTX!600
 2956   024240  000000                          0
 2957   024242  130000                          RELATP
 2958   024244  057600                          57600
 2959   024246  114000                          POINT
 2960   024250  000020                          20
 2961   024252  001100                          1100
 2962   024254  102200                          CHAR!INT1                       ;CHAR MODE
 2963   024256  036531  030461  030060          .ASCII  /Y=1100/
 2964   024264  114000                          POINT
 2965   024266  000200                          200
 2966   024270  001100                          1100
 2967   024272  110000                          LONGV                   ;DRAW HORIZ. LINE
 2968   024274  040600                          INTX!600
 2969   024276  000000                          0
 2970   024300  130000                          RELATP
 2971   024302  057600                          57600
 2972   024304  114000                          POINT
 2973   024306  000020                          20
 2974   024310  001000                          1000
 2975   024312  102000                          CHAR!INT0                       ;CHAR MODE
 2976   024314  036531  030061  030060          .ASCII  /Y=1000/
 2977   024322  114000                          POINT
 2978   024324  000200                          200
 2979   024326  001000                          1000
 2980   024330  110000                          LONGV                   ;DRAW HORIZ. LINE
 2981   024332  040600                          INTX!600
```

```
 2982  024334  000000                          0
 2983  024336  130000                          RELATP
 2984  024340  057600                          57600
 2985
 2986                                          .SBTTL  DRAW OUTER REFERENCE BOX
 2987
 2988  024342  117000                          POINT!INT4
 2989  024344  000000                          0
 2990  024346  000000                          0
 2991  024350  110000                          LONGV
 2992  024352  041777                          INTX!MAXX
 2993  024354  000000                          0
 2994  024356  040000                          INTX
 2995  024360  001777                          MAXY
 2996  024362  061777                          INTX!MINUSX!MAXX
 2997  024364  000000                          0
 2998  024366  040000                          INTX
 2999  024370  021777                          MINUSX!MAXY
 3000  024372  173400                          DSTOP
 3001  024374  160000                          DJMP
 3002  024376  022040                          FRME16
 3003
 3004                                          .SBTTL
 3005                                          .SBTTL  KEYBOARD CHARACTER ECHO SUB-PICTURE
 3006  024400  114000                ECHOFR:   POINT
 3007  024402  000000                          0
 3008  024404  001577                          MAXY-200
 3009  024406  170010                          STATSA!SYNC40                   ;ENABLE SYNC
 3010  024410  154240                          CHARS1                          ;ENABLE NORMAL CHAR. SIZE
 3011  024412  100000                          CHAR
 3012  024414     017     017                  .BYTE   17,17
 3013  024416  042513  041131  040517          .ASCII  /KEYBOARD CHARACTER ECHO LOOP/<15><12>
       024424  042122  041440  040510
       024432  040522  052103  051105
       024440  042440  044103  020117
       024446  047514  050117  005015
 3014  024454  020040  052103  046122          .ASCII  /  CTRL C TO EXIT LOOP/<15><12><12>
       024462  041440  052040  020117
       024470  054105  052111  046040
       024476  047517  006520  005012
 3015  024504  044103  051101  041501          .ASCII  /CHARACTER CODE  IS  = /
       024512  042524  020122  047503
       024520  042504  020040  051511
       024526  020040  020075
 3016  024532     000     000     000          .BYTE   0,0,0,0                 ;OCTAL VALUE CODE IS LOADED HERE
       024535     000
 3017  024536     015     012          ECODEV:  .BYTE   15,12
 3018  024540  161010                ECHJMP:   DJMPR!10                ;BR OVER IF NOT "SHIFTOUT" MODE
 3019  024542  100000                          CHAR
 3020  024544  044123  043111  026524          .ASCII  /SHIFT-OUT MODE/
       024552  052517  020124  047515
       024560  042504
 3021  024562     015     012                  .BYTE   15,12
 3022  024564  160000                          DJMP
```

```
                                        BUFFER

                                        .SBTTL
                                        .SBTTL  DYNAMIC EXT. STOP FRAME
                                        .SBTTL

                                ;DISPLAY A BOX AROUND THE SCREEN
                                ;       EACH LINE IS A DIFFERENT LINE TYPE AND INTENSITY LEVEL

024570  164300          FRME17: CONSL0!BIT7!BIT6                ;ENABLE CONSOLE #0
024572  164700                  CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
024574  114000                  POINT                           ;POINT
024576  000000                  0
024600  001777                  MAXY                            ;TO TOP LEFT CORNOR

024602  150001                  DNAME!BIT0                      ;LOAD NAME REG. WITH #1

024604  113407                  LONGV!INT6!LINE3                ;LONG VECTOR WITH INTENS. 6 AND LINE TYPE 3
024606  041777                  INTX!MAXX
024610  000000                  0

024612  113006                  LONGV!INT4!LINE2                ;LONG VECTOR WITH INTENS. 4 AND LINE TYPE 2
024614  040000                  INTX
024616  021777                  MINUSX!MAXY

024620  112405                  LONGV!INT2!LINE1                ;LONG VECTOR WITH INTENS. 2 AND LINE TYPE 1
024622  061777                  INTX!MINUSX!MAXX
024624  000000                  0

024626  113604                  LONGV!INT7!LINE0                ;LONG VECTOR WITH INTENS. 7 AND LINE TYPE 0
024630  040000                  INTX
024632  001777                  MAXY
```

```
3057                                    ;DISPLAY A DIAMOND -- WITH SHORT VECTORS AND DIFFERENT INTENSITY LEVELS
3058
3059  024634  150004                            DNAME!BIT2                      ;LOAD NAME REG. WITH BIT 2
3060  024636  114000                            POINT
3061  024640  001000                            1000
3062  024642  001500                            1500
3063  024644  106200                            SHORTV!INT1
3064  024646  057677                            57677                           ;+X +Y
3065  024650  106600                            SHORTV!INT3
3066  024652  077677                            77677                           ;+X -Y
3067  024654  107200                            SHORTV!INT5
3068  024656  077777                            77777                           ;-X -Y
3069  024660  107600                            SHORTV!INT7
3070  024662  057777                            57777                           ;-X +Y
3071
3072                                    ;DISPLAY FOUR BLINKING POINTS -- WITH RELATIVE POINT AND BLINK ENABLED
3073
3074  024664  150010                            DNAME!BIT3                      ;LOAD NAME REG. WITH #10
3075  024666  114000                            POINT
3076  024670  001000                            1000
3077  024672  000700                            700
3078  024674  133030                            RELATP!INT4!BLKON               ;RELATIVE POINT AND BLINK ON
3079  024676  057677                            57677                           ;+X +Y
3080  024700  077677                            77677                           ;+X -Y
3081  024702  077777                            77777                           ;-X -Y
3082  024704  057777                            57777                           ;-X +Y
3083
3084                                    ;DISPLAY FIVE GRAPHPLOT X DATA POINTS
3085
3086  024706  150020                            DNAME!BIT4                      ;LOAD NAME REG. WITH BIT4
3087  024710  174110                            STATSB!INCR+10                  ;LOAD GRAPH INCREMENT
3088  024712  114020                            POINT!BLKOFF
3089  024714  001000                            1000
3090  024716  001600                            1600
3091
3092  024720  120000                            GRAPHX
3093  024722  001500  001510  001520            1500,   1510,   1520,   1530,   1540
      024730  001530  001540
3094
3095                                    ;DISPLAY FIVE GRAPHPLOT Y DATA POINTS
3096
3097  024734  150040                            DNAME!BIT5                      ;LOAD NAME REG. WITH BIT5
3098  024736  114000                            POINT
3099  024740  001540                            1540
3100  024742  001200                            1200
3101
3102  024744  124000                            GRAPHY
3103  024746  001640  001630  001620            1640,   1630,   1620,   1610
      024754  001610
3104
```

```
3106                                    ;DISPLAY AN OCTOGON -- USING BASIC VECTOR'S
3107
3108  024756  150060                            DNAME!BIT5!BIT4                 ;LOAD NAME REG. WITH #60
3109  024760  114000                            POINT
3110  024762  001540                            1540
3111  024764  000640                            640
3112
3113  024766  120000                            BASICV
3114  024770  042100                            INTX!PATH0!100
3115  024772  046100                            INTX!PATH1!100
3116  024774  052100                            INTX!PATH2!100
3117  024776  056100                            INTX!PATH3!100
3118  025000  062100                            INTX!PATH4!100
3119  025002  066100                            INTX!PATH5!100
3120  025004  072100                            INTX!PATH6!100
3121  025006  076100                            INTX!PATH7!100
3122
3123                                    ;DISPLAY A LARGE SQUARE IN THE CENTER -- USING ABSOLUTE VECTORS
3124
3125  025010  150100                            DNAME!BIT6                      ;LOAD NAME REG. WITH BIT6
3126  025012  114000                            POINT
3127  025014  000400                            400
3128  025016  000400                            400
3129
3130  025020  144000                            ABSVCT
3131  025022  041400                            INTX!1400
3132  025024  000400                            400
3133
3134  025026  041400                            INTX!1400
3135  025030  001400                            1400
3136
3137  025032  040400                            INTX!400
3138  025034  001400                            1400
3139
3140  025036  040400                            INTX!400
3141  025040  000400                            400
3142
```

```
                                        ;NOW USE CHAR MODE AND DISP. JSR'S, CHAR. ROTATE, CHAR ITALICS

025042  150400                                DNAME!BIT8                      ;LOAD NAME REG. WITH BIT8
025044  170060                                STATSA!ITAL1                    ;ITALICS ON
025046  155400                                CHRRT1                          ;CHAR. ROTATE ON

025050  162000                                DJSR                            ;ABSOLUTE JSR TO CHAR. FRAME
025052  025066                                SHOWCH

                                        ;NOW USE CHAR MODE, DISP. JSR'S

025054  151000                                DNAME!BIT9                      ;LOAD NAME REG. WITH BIT9
025056  170040                                STATSA!ITAL0                    ;ITALICS OFF
025060  155000                                CHRRT0                          ;CHAR. ROTATE OFF

025062  163001                                DJSRR!1                         ;RELATIVE DJSR TO CHAR. FRAME
025064  161036                                DJMPR!36                        ;RELATIVE DJUMP OVER CHAR. SUBROUTINE

025066  114000                        SHOWCH: POINT
025070  000200                                200
025072  000200                                200

025074  154200                                CHARS0                          ;SET CHAR. SIZE TO 00
025076  100000                                CHAR
025100  020040  027060  020065                .ASCII  "  0.5 SIZE"
025106  044523  042532

025112  154240                                CHARS1                          ;SET CHAR. SIZE TO 01
025114  020040  027061  020060                .ASCII  "  1.0 SIZE"
025122  044523  042532

025126  154300                                CHARS2                          ;SET CHAR. SIZE TO 10
025130  020040  027061  020065                .ASCII  "  1.5 SIZE"
025136  044523  042532

025142  154340                                CHARS3                          ;SET CHAR. SIZE TO 11
025144  020040  027062  020060                .ASCII  "  2.0 SIZE"
025152  044523  042532

025156  154240                                CHARS1                          ;RESET CHAR. SIZE TO NORMAL
025160  166000                                DPOP                            ;EXIT
```

```
3182                                    ;DISPLAY A RECTANGLE IN THE MENU AREA -- USE DIFFERENT VECTOR SCALES
3183
3184  025162  151400                    $FILE2: DNAME!BIT9!BIT8                 ;LOAD NAME REG. WITH #1400
3185  025164  170003                            DMENU1                          ;ENABLE THE MENU AREA
3186  025166  114000                            POINT
3187  025170  000000                            0
3188  025172  000040                            40
3189
3190  025174  154037                            VCTR00!17                       ;LOAD VECTOR SCALE
3191  025176  110000                            LONGV
3192  025200  040000                            INTX                            ;DRAW VERT. LINE
3193  025202  000400                            400
3194
3195  025204  154021                            VCTR00!1                        ;LOAD VECTOR SCALE
3196  025206  040700                            INTX!700
3197  025210  000000                            0
3198
3199  025212  154037                            VCTR00!17                       ;LOAD VECTOR SCALE
3200  025214  040000                            INTX
3201  025216  020400                            MINUSX!400                      ;DRAW VERT. LINE
3202
3203  025220  154021                            VCTR00!1                        ;LOAD VECTOR SCALE
3204  025222  060700                            INTX!MINUSX!700
3205  025224  000000                            0
3206
3207  025226  170040                            STATSA!ITALO                    ;DISABLE ITALICS
3208  025230  154024                            VCTR00!4                        ;RETURN TO NORMAL SCALE
3209  025232  170002                            DMENU0                          ;EXIT MENU AREA
3210
3211  025234  173400                            DSTOP
3212
3213  025236  160000                            DJMP                            ;JUMP TO START OF FILE
3214  025240  024570                    FRM17F: FRME17
3215
3216
3217  025242  164700                    FRM17E: CONSL1!BIT7!BIT6                ;ENABLE CONSOLE #1
3218  025244  117030                            POINT!INT4!BLKON
3219  025246  000000                            0
3220  025250  001000                            1000
3221  025252  170040                            STATSA!ITALO                    ;ITALICS OFF
3222  025254  155000                            CHRRT0                          ;CHAR. ROT. OFF
3223  025256  154340                            CHARS3
3224
3225  025260  100000                            CHAR
3226  025262  162000                            DJSR                            ;JSR TO ASCII ERROR MESSAGE
3227  025264  025276                    WHY:    WHY0                            ;ADDRESS OF ERROR TYPE
3228  025266  100020                            CHAR!BLKOFF
3229  025270  173400                            DSTOP
3230  025272  160000                            DJMP
3231  025274  025242                            FRM17E
3232
```

```
 3234  025276  047516  042440  052130  WHY0:   .ASCII  /NO EXTERNAL STOP INTERRUPT/
       025304  051105  040516  020114
       025312  052123  050117  044440
       025320  052116  051105  052522
       025326  052120
 3235  025330  166000                          DPOP
 3236
 3237  025332  047125  054105  042520  WHY1:   .ASCII  /UNEXPECTED INTERRUPT TO VECTOR +4 /
       025340  052103  042105  044440
       025346  052116  051105  052522
       025354  052120  052040  020117
       025362  042526  052103  051117
       025370  025440  020064
 3238  025374  166000                          DPOP
 3239
 3240  025376  047125  054105  042520  WHY2:   .ASCII  /UNEXPECTED INTERRUPT TO VECTOR +10/
       025404  052103  042105  044440
       025412  052116  051105  052522
       025420  052120  052040  020117
       025426  042526  052103  051117
       025434  025440  030061
 3241  025440  166000                          DPOP
 3242
 3243  025442  047125  054105  042520  WHY3:   .ASCII  /UNEXPECTED INTERRUPT TO VECTOR +14/
       025450  052103  042105  044440
       025456  052116  051105  052522
       025464  052120  052040  020117
       025472  042526  052103  051117
       025500  025440  032061
 3244  025504  166000                          DPOP
 3245
 3246  025506  027104  027120  027103  WHY4:   .ASCII  /D.P.C. TOO LOW/
       025514  052040  047517  046040
       025522  053517
 3247  025524  166000                          DPOP
 3248
 3249  025526  027104  027120  027103  WHY5:   .ASCII  /D.P.C. TOO HIGH /
       025534  052040  047517  044040
       025542  043511  020110
 3250  025546  166000                          DPOP
 3251
 3252  025550  052123  050117  044440  WHY6:   .ASCII  /STOP INTERRUPT BUT NO STOP FLAGS/
       025556  052116  051105  052522
       025564  052120  041040  052125
       025572  047040  020117  052123
       025600  050117  043040  040514
       025606  051507
 3253  025610  166000                          DPOP
```

```
                                .SBTTL  SCOPE HANDLER ROUTINE
                                ;;*****************************************************************
                                ;*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
                                ;*AND LOAD THE TEST NUMBER($TSTNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
                                ;*AND LOAD THE ERROR FLAG ($ERFLG) INTO DISPLAY<15:08>
                                ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
                                ;*SW14=1         LOOP ON TEST
                                ;*SW08=1         LOOP ON TEST IN SWR<7:0>
                                ;*CALL
                                ;*      SCOPE           ;;SCOPE=IOT
                                
025612                          $SCOPE:
025612  032777  040000  153320          BIT     #BIT14,@SWR             ;TEST IF SW14 = 1
025620  001047                        BNE     $OVER                   ;BR IF SET
025622  005737  007506                TST     HOLD                    ;TEST IF LOOP ON PICTURE ?
025626  001044                        BNE     $OVER                   ;BR IF LOOP ON THIS TEST
025630  032777  040000  153302  1$:   BIT     #BIT14,@SWR     ;;LOOP ON PRESENT TEST?
025636  001040                        BNE     $OVER           ;;YES IF SW14=1
                                ;#####START OF CODE FOR THE XOR TESTER#####
025640  000416                  $XTSTR: BR      6$              ;;IF RUNNING ON THE "XOR" TESTER CHANGE
                                                                ;;THIS INSTRUCTION TO A "NOP" (NOP=240)
025642  013746  000004                  MOV     @#ERRVEC,-(SP)  ;;SAVE THE CONTENTS OF THE ERROR VECTOR
025646  012737  025666  000004          MOV     #5$,@#ERRVEC    ;;SET FOR TIMEOUT
025654  005737  177060                TST     @#177060        ;;TIME OUT ON XOR?
025660  012637  000004                MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
025664  000414                        BR      $SVLAD          ;;GO TO THE NEXT TEST
025666  022626                  5$:   CMP     (SP)+,(SP)+     ;;CLEAR THE STACK AFTER A TIME OUT
025670  012637  000004                MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
025674  000421                        BR      $OVER           ;;LOOP ON THE PRESENT TEST
025676                          6$:;#####END OF CODE FOR THE XOR TESTER#####
025676  032777  000400  153234          BIT     #BIT08,@SWR     ;;LOOP ON SPEC. TEST?
025704  001404                        BEQ     $SVLAD          ;;BR IF NO
025706  127737  153226  001102          CMPB    @SWR,$TSTNM     ;;ON THE RIGHT TEST?    SWR<7:0>
025714  001411                        BEQ     $OVER           ;;BR IF YES
025716  105237  001102          $SVLAD: INCB    $TSTNM          ;;COUNT TEST NUMBERS
025722  113737  001102  001176        MOVB    $TSTNM,$TESTN   ;;SET TEST NUMBER IN APT MAILBOX
025730  011637  001106                MOV     (SP),$LPADR     ;;SAVE SCOPE LOOP ADDRESS
025734  105037  001103                CLRB    $ERFLG          ;;ZERO THE ERROR FLAG
025740  013777  001102  153174  $OVER: MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER
025746  013716  001106                MOV     $LPADR,(SP)     ;;FUDGE RETURN ADDRESS
025752  000002                        RTI                     ;;FIXES PS
025752                                  .=.-2
025752  000005                        RESET
025754  005737  002230                TST     KRBD                    ;TEST IF KEYBOARD CONTROL
025760  001403                        BEQ     1$                      ;BR IF NOT
025762  052777  000100  153154        BIS     #BIT6,@$TKS             ;ENABLE KEYBOARD INTR.
025770  000002                  1$:   RTI
025772  000240                        NOP
025774  000240                        NOP
000200                                  APTSIZE=200
```

```
                                        ;DISPATCH TABLE OF THE STARTING ADDRESSES OF EACH TEST
                                        
026776  002232                  DISPTC: TST1
026000  002244                          TST2
026002  002256                          TST3
026004  002452                          TST4
026006  002500                          TST5
026010  002536                          TST6
026012  003140                          TST7
026014  003354                          TST10
026016  003366                          TST11
026020  003442                          TST12
026022  003704                          TST13
026024  003732                          TST14
026026  004016                          TST15
026030  004526                          TST16
026032  004670                          TST17
026034  004702                          TST20
026036  005046                          TST21
026040  005132                          TST22
026042  005256                          TST23
026044  005530                          TST24
026046  005564                          TST25
026050  006514                          TST27
026052  005776                          TST26
026054  000000                          0
026056  000000                          0
026060  000000                          0
026062  000000                          0
026064  000000                          0
026066  000000                          0

026070  000000                  BUFFER: 0

        000001                          .END
```

```
ABASE = 172000            14#    24
ABSVCT= 144000           164#  1718    1774    1829    1895    2122    3130
ACDW1 = 000000            24
ACDW2 = 000000            24
ACPUOP= 000000            24
ACRLF   004254           619    627#
ADDW0 = 000000            24
ADDW1 = 000000            24
ADDW10= 000000            24
ADDW11= 000000            24
ADDW12= 000000            24
ADDW13= 000000            24
ADDW14= 000000            24
ADDW15= 000000            24
ADDW2 = 000000            24
ADDW3 = 000000            24
ADDW4 = 000000            24
ADDW5 = 000000            24
ADDW6 = 000000            24
ADDW7 = 000000            24
ADDW8 = 000000            24
ADDW9 = 000000            24
ADEVCT= 000000            24
ADEVM = 000000            24
AENV  = 000000            24
AENVM = 000000            24
AFATAL= 000000            24
AMADR1= 000000            24
AMADR2= 000000            24
AMADR3= 000000            24
AMADR4= 000000            24
AMAMS1= 000000            24
AMAMS2= 000000            24
AMAMS3= 000000            24
AMAMS4= 000000            24
AMSGAD= 000000            24
AMSGLG= 000000            24
AMSGTY= 000000            24
AMTYP1= 000000            24
AMTYP2= 000000            24
AMTYP3= 000000            24
AMTYP4= 000000            24
APASS = 000000            24
APRIOR= 000200            16#    24
APTSIZ= 000200            69   3270#
ASWREG= 000000            24
ATESTN= 000000            24
AUNIT = 000000            24
AUSWR = 000000            24
AVECT1= 100320            15#    24
AVECT2= 000000            24
BADDON  006356           961    964     967     970     972     974     978#
BADO    006262           943    960#
BAD1    006272           911    962#
```

BAD2 006304 913 965#
BAD3 006316 915 968#
BAD4 006330 935 971#
BAD5 006340 938 973#
BAD6 006350 947 975#
BASICS= 134000 163# 1916 1925
BASICV= 120000 161# 411 1643 1648 1653 1658 1663 1668 1673 1678 1942 2007 2134
2151 2167 2646 3113
BEGIN 001334 19 68# 121 921
BIT0 = 000001 13# 213 240 952 1361 1362 1383 1420 3038
BIT00 = 000001 13#
BIT01 = 000002 13#
BIT02 = 000004 13#
BIT03 = 000010 13#
BIT04 = 000020 13#
BIT05 = 000040 13#
BIT06 = 000100 13#
BIT07 = 000200 13#
BIT08 = 000400 13# 3261
BIT09 = 001000 13#
BIT1 = 000002 13# 212 239 1359 1360 1381 1418
BIT10 = 002000 13# 201 205 237
BIT11 = 004000 13#
BIT12 = 010000 13# 500 501 508 512 513 520 524 525 532 536 537 544
931 1092 1936 1937
BIT13 = 020000 13# 1096
BIT14 = 040000 13# 248 985 1104 3261
BIT15 = 100000 13# 1231
BIT2 = 000004 13# 1357 1358 1379 1416 2709 2710 3059
BIT3 = 000010 13# 336 1355 1356 1377 1414 2709 2710 3074
BIT4 = 000020 13# 31 235 243 1353 1354 1375 1412 2709 2710 2735 2757 3086
3108
BIT5 = 000040 13# 231 233 242 1351 1352 1373 1410 2709 2710 2716 2735 2736
2757 3097 3108
BIT6 = 000100 13# 232 233 285 359 434 559 580 1021 1084 1266 1328 1334
1341 1349 1350 1371 1408 1429 1691 1938 2003 2044 2108 2181 2228
2247 2298 2318 2531 2709 2710 2735 2757 3032 3033 3125 3217 3266
BIT7 = 000200 13# 230 359 434 548 559 580 930 1053 1088 1266 1321 1327
1328 1334 1341 1347 1348 1369 1406 1429 1691 1938 2003 2044 2108
2181 2228 2247 2298 2318 2531 2709 2710 2716 2735 2736 2757 3032
3033 3217
BIT8 = 000400 13# 208 228 481 1112 1345 1346 1367 1404 2100 3146 3184
BIT9 = 001000 13# 200 202 206 227 345 1190 1343 1344 3155 3194
BLKOFF= 000020 182# 2055 2066 2077 2088 3088 3228
BLKON = 000030 183# 2052 2063 2074 2085 3078 3218
BPTVEC= 000014 13#
BUFFER 026070 358 422 428 433 461 472 558 573 578 598 654* 655* 658
660* 663 677* 678* 679* 682 684* 687 738 809 819 854 875
883 896 924 1008 1017 1148* 2124 2666 3023 3305#
CHAQU 020502 552 2531# 2628
CHAR = 100000 155# 584 1267 1325 1332 1975 2016 2033 2045 2055 2066 2077 2541
2554 2567 2575 2613 2624 2671 2686 2720 2729 2740 2749 2758 2827
2846 2867 2872 2877 2884 2897 2910 2923 2936 2949 2962 2975 3011
3019 3167 3225 3228

```
CHARQA  021240          565    2657   2670#
CHARQD  021536         2685#
CHARS0= 154200          230#    231    232    233   2541   2554   2567   2575   2613   2624   3166
CHARS1= 154240          231#   1981   2043   2541   2554   2567   2575   2613   2624   2625   3010   3170   3178
CHARS2= 154300          232#   2541   2554   2567   2575   2613   2624   3173
CHARS3= 154340          233#   1973   2541   2554   2567   2575   2613   2624   3176   3223
CHRRT0= 155000          227#    228    579    654   2662   3157   3222
CHRRT1= 155400          228#    677   2654   3148
CNTR    007512          822*   825*  1068*  1073*  1163#
CONSL0= 164000          207#   1327   1334   2710   2736   2757   3032
CONSL1= 164400          208#    359    434    559    580   1266   1321   1328   1341   1429   1691   1938   2003
                       2044   2108   2181   2228   2247   2298   2318   2531   2709   2716   2735   3033   3217
COUNT   007616         1173*  1183*  1194#
CR    = 000015           13#
CRLF  = 000200           13#
DDISP = 177570           13#     24     69
DDONE   001314           54#     74     76     81     97*   983*
DDONE1  001316           55#    100*
DELAY   001254           31#    325
DELTX6  007700          739*   747*   748    759   1217#
DELTX7  015752          766*   778*   779   2132#
DELTY6  007702          740*   742*   743    745*   760   1218#
DELTY7  016232          783*   804*   805   2149#
DELT11  016512          782*   801    803*  2166#
DFI10A  015756         2134#  2143
DFI11C  016236         2151#  2160
DIDINT  007472          926*   939    953*  1154#
DISPLA  001142           24#    69*  3261*
DISPRE  000174           19#    69
DISPTC  025776          137    140   3274#
DJMP  = 160000          199#   200    201    421    460    568    597    753    818    882   1013   1336   1423
                       1682   1931   1948   1962   1983   2021   2035   2123   2142   2159   2175   2192   2209
                       2239   2278   2303   2313   2411   2523   2627   2665   2730   2750   2760   2764   2767
                       2770   2773   3001   3022   3213   3230
DJMPR = 161000          200#   2100   3018   3160
DJSR  = 162000          201#    202    564   1976   1979   2017   2257   2259   2273   2275   2582   2585   2588
                       2591   3150   3226
DJSRR = 163000          202#   2661   3159
DLT14A  022130          107*   108*   1107   2722#
DLT14B  022142          109*   110*   1110   2725#
DLT14C  022200          111*   112*   1115   2742#
DLT14D  022212          113*   114*   1118   2745#
DMENU0= 170002          212#    213   1526   2150   2663   3209
DMENU1= 170003          213#   1493   2195   2642   3185
DNAME = 150000          244#   3038   3059   3074   3086   3097   3108   3125   3146   3155   3184
DNOP  = 164000          204#    205    206    207    208   1037   1764   1765   1766   1767   1768   1820   1821
                       1822   1823   1875   1876   1877   1878   2594   2595   2596   2597   2598   2599   2600
DPC     001262           40#     70    919*   932    948*   952*   978   1052   1121*  1175*  1185*
DPOP  = 166000          205#   1997   2294   2636   2701   3179   3235   3238   3241   3244   3247   3250   3253
DPOPNR= 165000          206#
DSAVE   007476          876*   898*  1157#
DSAVE1  007500          872*   900*  1158#
DSAVE2  007502         1079*  1081*  1106   1114   1132   1159#
DSAVE3  007504         1080*  1082*  1109   1117   1140   1160#
```

DZVSDB.P11 CROSS REFERENCE TABLE.

```
DSR     001264          41#    946     979
DSREL   001272          44#
DSR1    001274          45#    930*    944     980
DSTOP = 173400         211#    420     459     596     752     817     881    1335    1422    1681    1930    1947    1982
                      2020    2091    2141    2158    2174    2191    2208    2238    2277    2302    2312    2410    2522
                      2626    2664    3000    3211    3229
DSWR  = 177570          13#     24      69
ECHJMP  024540        1015*   1037*   1043*   3018#
ECHOFR  024400        1014    1024    3006#
ECODEV  024536        1006*   1007*   1048    3017#
EDGE0 = 176040         242#    243
EDGE1 = 176060         243#
EMTVEC= 000030          13#
ERRVEC= 000004          13#     69*    3261*
FILE    007620        1174*   1175    1195#
FILE4A  004332         599     653#    672
FILLIT  004276         614     618     635#
FIL14A  005364         820     830#
FIXVCT  001664          95      97#    136
FRME0   010006         274    1263#   1337
FRME10  015750         770    2131#
FRME11  016226         787    2147#
FRME14  017064         835     840     845     850    2228#   2240
FRME16  022040         891    2709#   3002
FRME17  024570         919     933     948    3032#   3214
FRME2   012306         339     348    1429#   1683
FRME3   013336         478    1688#   1932
FRME5   015450         735    2040#   2094
FRME6   015676         754     757    2104#
FRM1C   016764         773     790    2178#   2193
FRM11D  016514        2167#   2176
FRM11M  017024         793    2195#   2210
FRM11S  016506         796    2164#
FRM14A  017070         831*    836*    841*    846*   2230#
FRM14B  017072         832*    837*    842*    847*   2231#
FRM16B  022264         885*    886*   1128    2760#
FRM16C  022430        2761    2786#
FRM17E  025242         986    3217#   3231
FRM17F  025240         936    3214#
GNS   = ****** U        19
GRAPH   017176         863    2244#   2279
GRAPHX= 120000         159#    161    2272    3092
GRAPHY= 124000         160#   2256    3102
GRPINC  017222         860*    864*    865    2255#
HALFX = 000777         252#    412     413     414     415     416     417     418     419
HITCNT  007474         884*   1126*   1127    1156#
HOLD    007506          84*    134*    143*    145*   1161#   3261
HT    = 000011          13#
INCR  = 000100         224#    860     865    2255    3087
INTX  = 040000         248#    296     367     381     394     405     412     413     414     415     416     417     418
                       419     437     445     449     457     739     748     823    1062    1148    1343    1345    1347
                      1349    1351    1353    1355    1357    1359    1361    1363    1366    1368    1370    1372    1374
                      1376    1378    1380    1382    1385    1387    1389    1391    1393    1395    1397    1399    1401
                      1404    1406    1408    1410    1412    1414    1416    1418    1420    1432    1434    1436    1438
```

```
                    1440    1442    1444    1446    1448    1450    1452    1454    1461    1463    1465    1467    1470
                    1472    1474    1476    1479    1481    1487    1489    1498    1500    1502    1504    1507    1509
                    1511    1513    1516    1518    1522    1524    1533    1535    1537    1539    1541    1543    1545
                    1547    1549    1551    1555    1557    1564    1566    1568    1570    1572    1574    1576    1578
                    1580    1582    1586    1588    1596    1602    1608    1614    1631    1644    1649    1654    1659
                    1664    1669    1674    1679    1711    1719    1720    1721    1722    1723    1724    1725    1726
                    1727    1728    1729    1730    1731    1732    1733    1734    1735    1736    1737    1738    1739
                    1740    1741    1742    1743    1744    1745    1746    1747    1748    1749    1750    1751    1752
                    1753    1754    1755    1756    1757    1758    1759    1760    1761    1762    1763    1775    1776
                    1777    1778    1779    1780    1781    1782    1783    1784    1785    1786    1787    1788    1789
                    1790    1791    1792    1793    1794    1795    1796    1797    1798    1799    1800    1801    1802
                    1803    1804    1805    1806    1807    1808    1809    1810    1811    1812    1813    1814    1815
                    1816    1817    1818    1819    1830    1831    1832    1833    1834    1835    1836    1837    1838
                    1839    1840    1841    1842    1843    1844    1845    1846    1847    1848    1849    1850    1851
                    1852    1853    1854    1855    1856    1857    1858    1859    1860    1861    1862    1863    1864
                    1865    1866    1867    1868    1869    1870    1871    1872    1873    1874    1882    1886    1890
                    1896    1898    1900    1902    1904    1906    1908    1910    1943    1944    1945    1946    1957
                    1968    2008    2009    2010    2011    2110    2116    2140    2157    2173    2183    2185    2187
                    2189    2200    2202    2204    2206    2233    2235    2237    2249    2266    2310    2324    2326
                    2328    2330    2341    2351    2361    2372    2382    2384    2386    2388    2401    2403    2405
                    2407    2419    2425    2431    2437    2443    2449    2455    2461    2471    2477    2485    2491
                    2499    2505    2513    2519    2536    2549    2562    2608    2619    2631    2632    2633    2634
                    2647    2648    2649    2650    2714    2790    2793    2794    2795    2796    2797    2798    2799
                    2800    2801    2833    2834    2835    2836    2837    2838    2839    2840    2861    2890    2903
                    2916    2929    2942    2955    2968    2981    2992    2994    2996    2998    3041    3045    3049
                    3053    3114    3115    3116    3117    3118    3119    3120    3121    3131    3134    3137    3140
                    3192    3196    3200    3204
INTO   = 002000      171#    2975
INT1   = 002200      172#    1939    2962    3063
INT2   = 002400      173#    2949    3048
INT3   = 002600      174#    2936    3065
INT4   = 003000      175#     464    2040    2299    2571    2754    2923    2988    3044    3078    3218
INT5   = 003200      176#    2910    3067
INT6   = 003400      177#    2897    3040
INT7   = 003600      178#     877    2109    2134    2151    2167    2178    2196    2394    2884    3052    3069
IOTVEC= 000020       13#      69*
ITAL0  = 000040      217#     591     612    1974    2670    2685    2700    3156    3207    3221
ITAL1  = 000060      218#     594     616    1978    2679    2697    3147
J      = 001000     2880#
KBCHR    007624     1049    1108    1111    1116    1119    1129    1200#
KRBD     002230       89*     92*     151#   3264
L      = 000025     2848#   2861#
LF     = 000012       13#
LINE0  = 000004      185#    2045    2047    2088    2244    3052
LINE1  = 000005      186#    2058    3048
LINE2  = 000006      187#    2069    3044
LINE3  = 000007      188#    2080    3040
LOADAC   006776     1060#   1069
LOADBF   004210      586     588     590     612#
LOADSP   004154      592     595     601#
LOADUP   007020      880    1068#
LOADVT   005336      814     816     822#
LONGV  = 110000      157#    404     467    1070    1460    1497    1532    1563    1595    1601    1607    1613    1711
                    1882    1886    1890    1956    1967    2047    2052    2058    2063    2069    2074    2080    2085
```

```
                    2088    2109    2115    2182    2199    2248    2265    2309    2323    2335    2345    2355    2366
                    2377    2381    2398    2418    2424    2430    2436    2442    2448    2454    2460    2470    2476
                    2484    2490    2498    2504    2512    2518    2535    2548    2561    2607    2618    2714    2789
                    2793    2794    2795    2796    2797    2798    2799    2800    2801    2833    2834    2835    2836
                    2837    2838    2839    2840    2861    2889    2902    2915    2928    2941    2954    2967    2980
                    2991    3040    3044    3048    3052    3191
LPDARK= 000300        216#
LPLITE= 000200        215#
LPOFF = 000100        180#
LPON  = 000140        181#    2711    2754
LPRTA   022162       2735#    2765    2768
LPRTC   022232       2754#    2771    2774
LPVCT   001320         57#     101*    869*    902*
LPVCT1  001322         58#     101     102*    870*    871*    902     903*
LQ6   = 000040       1639#    1644    1649    1654    1659    1664    1669    1674    1679
M     = 000100       2849#    2861#
MAXMUX= 000177        249#    2173    2200    2204
MAXX  = 001777        250#     252     365     405     449     451     457     739    1227    1229    1438    1440    1442
                    1444    1446    1448    1461    1465    1472    1476    1479    1481    1485    1487    1489    2106
                    2116    2157    2183    2187    2326    2330    2353    2364    2375    2395    2790    2992    2996
                    3041    3049
MAXY  = 001777        251#     395     403     406     438     440     446     562     583     655     743     745     812
                     842     847    1225    1227    1265    1445    1447    1449    1451    1453    1455    1464    1468
                    1471    1475    1480    1482    1488    1490    1501    1505    1508    1512    1517    1519    1523
                    1525    2111    2114    2140    2186    2190    2203    2207    2325    2329    2344    2354    2396
                    2648    2650    2995    2999    3008    3036    3046    3054
MESG    007520       272     278     292     298     303     309     315     337     346     349     426     470     476
                     485     502     514     526     538     550     571     656     661     665     680     685     689
                     707     716     725     733     755     768     771     785     788     791     794     833     838
                     843     848     852     861     889     894     984    1022    1173#
MESGA   007536      1122    1176#    1178    1186
MESGAA  007546        97     983    1182#
MESGBA  007614      1188    1191    1193#
MINSUY= 000100        255#    2233    2235    2237
MINUSX= 020000        253#     367     406     440     443     451     456     512     520     544    1072    1465    1470
                    1474    1475    1476    1481    1482    1487    1490    1502    1505    1507    1510    1511    1512
                    1513    1514    1518    1519    1522    1525    1537    1543    1545    1547    1551    1552    1555
                    1558    1568    1574    1576    1578    1582    1583    1586    1589    1614    1631    1711    1882
                    1886    1890    2187    2190    2204    2207    2233    2235    2237    2329    2330    2341    2351
                    2361    2372    2379    2384    2386    2387    2389    2399    2405    2408    2437    2443    2456
                    2461    2492    2499    2505    2506    2513    2519    2633    2714    2833    2834    2835    2836
                    2837    2838    2839    2840    2996    2999    3046    3049    3201    3204
MINUSY= 020000        254#     536    1468    1473    1477    1540    1542    1546    1548    1571    1573    1577    1579
                    1609    2400    2486    2500
MSOPEN  022160        873*    1094*    1098*    2731#
MS1PEN  022230        874*    1086*    1090*    2751#
NAMEVT  001330         63#     105*
NAMEV1  001332         64#     105     106*
NO    = 047516        700#     705
OFFST0= 010000        166#
OFFST1= 012000        167#
OFFST2= 014000        168#
OFFST3= 016000        169#
OFFTST  015034        504      516      528      540     1935#    1949
```

```
OFFT1   015036              500*   507*   508    512*   519*   520    524*   537*  1936#
OFFT2   015040              501*   513*   525*   531*   532    536*   543*   544   1937#
ORELPT  021154             2583   2586   2589   2592   2630#
PATH0 = 002000              190#    412    417   1644   1943   2140   2157   2173   2647   3114
PATH1 = 006000              191#   1649   2008   3115
PATH2 = 012000              192#    414    419   1654   1944   2140   2157   2173   2648   3116
PATH3 = 016000              193#   1659   2009   3117
PATH4 = 022000              194#    413    416   1664   1945   2157   2173   2649   3118
PATH5 = 026000              195#   1669   2010   3119
PATH6 = 032000              196#    415    418   1674   1946   2140   2650   3120
PATH7 = 036000              197#   1679   2011   3121
PC    =%000007               13#     95*    115*   136*   435*   447*   468*   586*   588*   590*   592*   593*   595*
                            610*   614*   615*   618*   619*   620*   631*   640*   648*   741*   750*   763*   814*
                            816*   827*   880*   997*  1049*  1065*  1069*  1075*  1108*  1111*  1116*  1119*  1129*
                           1200*  1202*  1204*  1206*  1208*  1215*
PCERR   006426              978*   990#
PENOF0  022270              873   1094   2731   2763#
PENOF1  022350              874   1086   2751   2769#
PENON0  022320             1098   2766#
PENON1  022400             1090   2772#
PICSCA  020134              482*    489*   2397#
PICSCS  017572              487   2318#  2412
PICST0  012372              360   1457#
PICST1  012422              362   1470#
PICVTA  017552              365*    379*   2307#
PICVTB  017554              366*    376*    380*    390*   2308#
PICVTC  017560              367*    377*    381*    391*   2310#
PICVTE  017564              372     386    2312#
PICVTL  017550              370     384    2306#   2314
PIRQ  = 177772               13#
PIRGVE= 000240               13#
POINT = 114000              158#    393    401    408    464    560    581    810    877   1263   1322   1329   1342
                           1431   1457   1494   1528   1560   1592   1598   1604   1610   1617   1640   1645   1650
                           1655   1660   1665   1670   1675   1688   1715   1771   1826   1879   1883   1887   1892
                           1913   1922   1935   1939   1953   1959   1964   1970   2004   2012   2030   2040   2105
                           2112   2118   2131   2147   2164   2178   2196   2229   2244   2252   2262   2269   2299
                           2306   2320   2332   2342   2352   2363   2374   2394   2415   2421   2427   2433   2439
                           2445   2451   2457   2467   2473   2481   2487   2495   2501   2509   2515   2532   2538
                           2545   2551   2558   2564   2571   2578   2604   2610   2615   2621   2643   2651   2711
                           2717   2726   2737   2746   2754   2786   2793   2794   2795   2796   2797   2798   2799
                           2800   2801   2824   2829   2843   2861   2864   2869   2874   2881   2886   2894   2899
                           2907   2912   2920   2925   2933   2938   2946   2951   2959   2964   2972   2977   2988
                           3006   3034   3060   3075   3088   3098   3109   3126   3162   3186   3218
PR0   = 000000               13#
PR1   = 000040               13#
PR2   = 000100               13#
PR3   = 000140               13#
PR4   = 000200               13#
PR5   = 000240               13#
PR6   = 000300               13#
PR7   = 000340               13#
PS    = 177776               13#
PSW   = 177776               13#    917*   1176*
PWRVEC= 000024               13#
```

```
RELATP= 130000   162#  163   815  1061  1631  2234  2630  2892  2905  2918  2931  2944  2957
                2970  2983  3078
RESTAT  001516    70#  997
RESVEC= 000010    13#
RETB    002026    93   117#
RET14   007056   869  1078#
ROTCHR  021172   569  2642#
R0     =%000000   13#   70*   72*   74   76*   79*   81   98*   99*  100  286*  288  289
               358*  359*  361*  371*  385*  393*  396*  397*  401*  402*  403*  404*  405*
               406*  408*  409*  410*  411*  412*  413*  414*  415*  416*  417*  418*  419*
               420*  421*  422*  433*  434*  437*  438*  439*  440*  443*  444*  445*  446*
               449*  450*  451*  452*  455*  456*  457*  458*  459*  460*  461*  464*  465*
               466*  467*  558*  559*  560*  561*  562*  564*  565*  568*  569*  578*  579*
               580*  581*  582*  583*  584*  591*  594*  596*  597*  598*  601*  603*  609*
               612*  616*  627*  628*  629*  630*  636*  645*  738*  752*  753*  754*  759*
               760*  761*  762*  809*  810*  811*  812*  813*  815*  817*  818*  819*  823*
               824*  875*  877*  878*  879*  881*  882*  883*  907*  908*  909*  910*  911*
               912*  913*  914*  915*  916*  921*  922   924   997* 1008* 1010* 1013* 1014*
              1017* 1047* 1061* 1062* 1070* 1071* 1072* 1132* 1133* 1134* 1135* 1138* 1140*
              1141* 1142* 1143* 1146*
R1     =%000001   13#   71*   72   73*   77*   78*   79   80*  360*  361  362  370*  371
               372   384*  385  386   394*  396   398*  436*  441*  448* 453*  563*  566*
               922*  923*  927* 1009* 1011* 1018* 1029* 1060* 1063* 1130* 1137* 1145* 1148*
R2     =%000002   13#  395*  397   399*  602*  603   604*  605   607   613*  617*  636   637*
               920*  950* 1048* 1052* 1053* 1107* 1110* 1115* 1118* 1128* 1131* 1201* 1203*
              1205* 1207*
R3     =%000003   13#  635*  638*  644*  646*  931*  941* 1027* 1028* 1031  1034  1040  1044
              1046* 1047  1106* 1109* 1114* 1117* 1127* 1209* 1210* 1211* 1212
R4     =%000004   13#   85*  129*  130   132*  137   139*  140   1201  1203  1205  1207  1212*
              1213* 1214*
R5     =%000005   13#  272*  278*  292*  298*  301*  303*  306*  309*  312*  315*  324   330*
               337*  346*  349*  426*  470*  476*  485*  502*  514*  526*  538*  550*  571*
               656*  661*  665*  680*  685*  689*  707*  716*  725*  733*  755*  768*  771*
               785*  788*  791*  794*  833*  838*  843*  848*  852*  861*  889*  894*  984*
              1022* 1173  1174  1193*
R6     =%000006   13#   69*
R7     =%000007   13#
SECDLY  002416   301   306   312   324#
SHIFTO  007472  1016* 1036* 1038  1042* 1153#
SHORTV= 104000   156#  813  1631  2232  2236  3063  3065  3067  3069
SHOWCH  025066  3151  3162#
SINE    017300  2258  2260  2274  2276  2285#
SP     =%000006   13#   69*  131*  955   963   966   969  1051  1120  1182  3261*
SPACE   004314   593   615   644#
SPRAY   005030   741   750   759#
SRERR   006430   979*  991#
SR1ERR  006432   980*  992#
STACK = 001100    13#   69   131
STATE = 176000   237#  239   242
STATSA= 170000   210#  212   591   594   612   616   706   715   724  1974  1978  2002  2670
              2679  2685  2697  2700  3009  3147  3156  3207  3221
STATSB= 174000   222#  237   860   865  2255  3087
STATSC= 154000   226#  227   230   235
STCHAR  004252   585*  587*  589*  613   617   622#
```

```
STDPIC  017532            294     300     305     311     317    2298*   2304
STDRA   017536            288*    296*   2300*
STDRB   017540            289*   2301*
STKLMT= 177774             13*
STRNG0= 176002            239*    240    2019
STRNG1= 176003            240*   2015
SUBOFF= 000024            259*   1994
SUBON = 000022            258*   1992
SUPC0   015120            691    1964*
SUPC1   015142           1963    1973*
SUPOFF= 000023            257*   1990
SUPON = 000021            256*   1988
SUPPIC  015072            667    1953*   1984
SUPSUB  015172           1977    1980    1988*
SWITCH  007622             83*    124*    133*    147*    505     517     529     541     671     695     710     719     728
                           774     797    1189*   1192*   1196*
SWR     001140             24*     69*     90     1187    1190    3261
SWREG   000176             19*     69
SW0   = 000001             13*
SW00  = 000001             13*
SW01  = 000002             13*
SW02  = 000004             13*
SW03  = 000010             13*
SW04  = 000020             13*
SW05  = 000040             13*
SW06  = 000100             13*
SW07  = 000200             13*
SW08  = 000400             13*
SW09  = 001000             13*
SWi   = 000002             13*
SW10  = 002000             13*
SW11  = 004000             13*
SW12  = 010000             13*
SW13  = 020000             13*
SW14  = 040000             13*
SW15  = 100000             13*
SW2   = 000004             13*
SW3   = 000010             13*
SW4   = 000020             13*
SW5   = 000040             13*
SW6   = 000100             13*
SW7   = 000200             13*
SW8   = 000400             13*
SW9   = 001000             13*
SYNC30= 000004            219*    724
SYNC40= 000010            220*    715    3009
SYNPIC  015226            706*    709     715*    718     724*    727    2002*   2022
SYNSPD  015320            705*    714*    723*   2025*
SYNTXT  015300           2018    2024*
S30   = 030063            702*    723
S40   = 030064            701*    714
TABA    007704            286    1223*
TABB    012044            280    1341*   1424
TBITVE= 000014             13*
```

```
TEMPA   007514           285*    320*    336*    341*    345*    353*    369*    374*    383*    389*    483*    490*    548*
                         554*    653*    669*    676*    693*    767*    776*    784*    799*    830*    855*   1164#
TEMPB   007516           481*    493*   1165#
TIMEVT  001324            60#    103*
TKBVCT  001256            32#     87*     93*   1019*
TKBVT1  001260            33#     87      88*     94*   1020*
TKVEC = 000060            13#
TMEVT1  001326            61#    103     104*
TPVEC = 000064            13#
TRAPVE= 000034            13#
TRTVEC= 000014            13#
TSAVE   007510            86*    117*    118*    119     122     125*    127     129     141     149*   1162#
TST1    002232            96     270#   3274
TST10   003354           475#   3281
TST11   003366           480#   3282
TST12   003442           497#   3283
TST13   003704           547#   3284
TST14   003732           557#   3285
TST15   004016           576#   3286
TST16   004526           699#   3287
TST17   004670           732#   3288
TST2    002244           277#   3275
TST20   004702           737#   3289
TST21   005046           758     765#   3290
TST22   005132           781#   3291
TST23   005256           808#   3292
TST24   005530           859#   3293
TST25   005564           868#   3294
TST26   005776           906#   3296
TST27   006514          1005#   3295
TST3    002256           284#   3276
TST4    002452           322     335#   3277
TST5    002500           344#   3278
TST6    002536           356#   3279
TST7    003140           432#   3280
VCTROO= 154020           235#    482    2104    2121    2319    2393    2397    2409    2577    2581    2584    2587    2590
                        2593    3190    3195    3199    3203    3208
VIEW    007472           888*    893*   1102    1155#
VSCONS  001304            49#   1078
VSTERM  001312            52#    704*    713*    722*
VSTRT   020174           351    2415#   2524
WHERE1= 000265          2094#   2095#   2096#   2097#   2098#   2100
WHERE2= 000005          2657#   2658#   2659#   2661
WHY     025264           960*    962*    965*    968*    971*    973*    975*   3227#
WHY0    025276           960    3227    3234#
WHY1    025332           962    3237#
WHY2    025376           965    3240#
WHY3    025442           968    3243#
WHY4    025506           971    3246#
WHY5    025526           973    3249#
WHY6    025550           975    3252#
XDOFF   001276            46#
XPOS    001266            42#   1079
XQB   = 000353          1637#   1641    1646    1651    1656    1661    1666    1671    1676
```

```
YDOFF   001300            47#
YPOS    001270            43#   1080
YQ6   = 001003          1638#   1642    1647    1652    1657    1662    1667    1672    1677
$APTHD  001000            23#
$AUTOB  001134            24#
$BASE   001246            24#     71
$BDADR  001122            24#
$BDDAT  001126            24#    932*    934     937    1078*   1084    1088    1092    1096    1104    1112
$CDW1   001252            24#
$CMTAG  001100            24#     69
$CM1  = 000002            24#
$CM2  = 000004            24#
$CM3  = 000002            24#
$CPUOP  001220            24#
$CRLF   001167            24#
$DEVCT  001202            24#
$DEVM   001250            24#
$DOAGN  006510           997#
$ENDAD  006500            21     997#
$ENDCT  006464            69     997#
$ENV    001212            24#
$ENVM   001213            24#     69
$EOP    006434           956     988     997#   1033
$EOPCT  006456            69*    997#
$ERFLG  001103            24#   3261*
$ERMAX  001115            24#   3261
$ERRPC  001116            24#
$ERRTB  001254            24#
$ERTTL  001112            24#
$ETABL  001212            24#
$ETEND  001254            23      24#
$FATAL  001174            24#
$FILE2  025162          3184#
$FILLC  001156            24#
$FILLS  001155            24#
$GDADR  001120            24#
$GDDAT  001124            24#    933*    934     936*    937
$GET42  006470           997#
$HD   = 000000            12
$HIBTS  001000            23#
$ICNT   001104            24#
$INTAG  001135            24#
$ITEMB  001114            24#
$LF     001170            24#
$LPADR  001106            24#     69*    425*   3261*
$LPERR  001110            24#
$MADR1  001224            24#
$MADR2  001230            24#
$MADR3  001234            24#
$MADR4  001240            24#
$MAIL   001172            23      24#     69    3261
$MAMS1  001222            24#
$MAMS2  001226            24#
$MAMS3  001232            24#
```

```
$MAMS4  001236            24#
$MBADR  001002            23#
$MSGAD  001206            24#
$MSGLG  001210            24#
$MSGTY  001172            24#
$MTYP1  001223            24#
$MTYP2  001227            24#
$MTYP3  001233            24#
$MTYP4  001237            24#
$NULL   001154            24#
$NWTST= 000001           270#   277#   284#   335#   344#   356#   432#   475#   480#   497#   547#   557#   576#
                         699#   732#   737#   765#   781#   808#   859#   868#   906#  1005#
$OVER   025740          3261#
$PASS   001200            24#    69*   997*
$PASTM  001006            23#
$QUES   001166            24#
$REGAD  001160            24#
$REGO   001162            24#
$REG1   001164            24#
$RTNAD  006512           997#
$SCOPE  025612            69   3261#
$SETUP= 000021            27#    69    997   3261
$STUP = 177777            27#
$SVLAD  025716          3261#
$SVPC = 000204            21#
$SWR  = 040400             5#    12    18    24    69   270   277   284   335   344   356   432   475
                         480   497   547   557   576   699   732   737   765   781   808   859   868
                         906   997  1005  3261
$SWREG  001214            24#    69
$SWRMK= 000000            18  3261
$TESTN  001176            24#  3261*
$TKB    001146            24#   117  1027
$TKS    001144            24#  1021* 3266*
$TN   = 000030             6#    12   270#   277#   284#   322   335#   344#   356#   432#   475#   480#   497#
                         547#   557#   576#   699#   732#   737#   758   765#   781#   808#   859#   868#   906#
                        1005#
$TPB    001152            24#
$TPFLG  001157            24#
$TPS    001150            24#
$TSTM   001004            23#
$TSTNM  001102            24#   130*   997*  3261*
$UNIT   001204            24#
$UNITM  001010            23#
$USWR   001216            24#
$VECT1  001242            24#    77     98    870   907
$VECT2  001244            24#
$XTSTR  025640          3261#
$$GET4= 000000           997#
$40CAT= ****** U        3261
.     = 026072            19#    21#    22#    23#    24#    69    997   2094   2657   3261   3262#
.$X   = 001000            23#
```

```
CHACCU    1232#  2541   2554   2567   2575   2613   2624
CHRMGS    2687#  2695   2698
CHRMSG    2672#  2677   2680
CLEARA     262#
COMMEN      13#
DSTEP       34#  1121   1185
ENDCOM      13#
ERROR       13#
ESCAPE      13#
GETPRI      13#
GETSWR      13#
MSCOPE    3255#  3261
MULT        13#
NEWTST      13#   270    277    284    335    344    356    432    475    480    497    547    557    576    699
             732   737    765    781    808    859    868    906   1005
OCTGN     2214#  2233   2235   2237
OCTGON    1692#  1711   1882   1886   1890   2714
POP         13#
PUSH        13#
REPORT      13#
SCOPE       13#   270    277    284    335    344    356    432    475    480    497    547    557    576    699
             732   737    765    781    808    859    868    906    997   1005
SETPRI      13#
SETUP       13#    69
SKIP        13#   322    758
SLASH       13#
SPACE       13#
STARS       13#    21     23     24    270    277    284    335    344    356    432    475    480    497    547
             557   576    699    732    737    765    781    808    859    868    906    997   1005   3261
SWRSU       13#    69#
THRLIN    2803#  2833   2834   2835   2836   2837   2838   2839   2840
TYPBIN      13#
TYPDEC      13#
TYPNAM      13#
TYPNUM      13#
TYPOCS      13#
TYPOCT      13#
TYPTXT      13#
VRTLN     2777#  2793   2794   2795   2796   2797   2798   2799   2800   2801
$$CMRE      24#
$$CMTM      24#
$$ESCA      13#
$$NEWT      13#   270    277    284    335    344    356    432    475    480    497    547    557    576    699
             732   737    765    781    808    859    868    906   1005
$$SETM      69#
$$SKIP      13#   322    758
.EQUAT       7#    13
.HEADE       7#    12
.SETUP       8#    27
.SWRHI       9#    18
.SWRLO      18#
.$ACTI      10#    21
.$APTB      10#    24#
.$APTH      10#    23
```

```
.$APTY      10#
.$CATC       7#      19
.$CMTA       7#      24
.$EOP        7#     997
.$PARM       8#
.$POWE       8#
.$SCOP       8#    3261
.$SWDO       8#
.$TRAP       9#
.$TRPT       9#
.$TYPD       9#
.$TYPE       9#
```

ADD 73 80 376 377 390 398 507 519 531 543 742 1137 1145 1214
ASL 132
ASR 1134 1135 1142 1143
BEQ 69 91 123 375 389 606 802 925 951 997 1030 1039 1085 1089 1093
1097 1105 1113 1136 1144 1184 1191 3261 3265
BGT 997
BHI 935
BIC 78 99 118 871 908 923 997 1028 1046 1081 1082 1148 1213
BIS 296 930 953 1021 1053 3266
BIT 1084 1088 1092 1096 1104 1112 1190 3261
BITB 69
BLE 744
BLO 938
BLT 749
BMI 126 128 291 945 1188
BNE 69 75 82 120 138 142 321 327 342 354 363 373 387 442 454
491 494 506 509 518 521 530 533 542 545 555 567 608 639 647
670 672 694 696 711 720 729 775 780 798 806 826 856 866 899
901 928 940 942 1012 1032 1035 1041 1064 1074 1101 1103 1139 1147 3261
BPL 329 400 777 800 947 1045
BR 69 150 318 322 378 392 746 751 758 820 943 961 964 967 970
972 974 988 1025 1054 1149 1178 1186 3261
CLR 69 83 84 85 86 88 89 102 104 106 124 133 134 139 145
561 582 602 678 761 762 766 782 783 884 888 903 917 926 997
1010 1016 1042 1083 1130 1131 1176 1189
CLRB 3261
CMP 69 74 81 119 122 127 141 362 372 386 508 520 532 544 605
607 743 748 779 801 805 865 924 934 937 955 963 966 969 1031
1034 1040 1051 1120 1182 3261
CMPB 1044 3261
COM 92 1192
DEC 320 326 328 341 353 374 388 441 453 489 490 493 554 566 638
646 669 693 776 799 825 855 898 900 927 941 950 997 1011 1029
1063 1073 1138 1146 1183
EMT 13
HALT 19
INC 604 637 778 803 804 864 893 997 1036 1087 1091 1095 1099 1126
INCB 3261
IOT 13
JMP 19 96 121 140 462 599 956 997 1033 1122
JSR 95 136 272 278 292 298 301 303 306 309 312 315 337 346 349
426 435 447 470 476 485 502 514 526 538 550 571 586 588 590
592 593 595 614 615 618 619 656 661 665 680 685 689 707 716
725 733 741 750 755 768 771 785 788 791 794 814 816 833 838
843 848 852 861 880 889 894 984 997 1022 1049 1069 1108 1111 1116
1119 1129 1200 1202 1204 1206
MOV 69 70 71 72 76 77 79 87 93 94 97 100 101 103 105
107 108 109 110 111 112 113 114 129 131 143 147 285 286 288
289 324 325 336 345 358 359 360 361 365 366 367 369 370 371
379 380 381 383 384 385 393 394 395 396 397 401 402 403 404
405 406 408 409 410 411 412 413 414 415 416 417 418 419 420
421 422 425 433 434 436 437 438 439 440 443 444 445 446 448
449 450 451 452 455 456 457 458 459 460 461 464 465 466 467
481 482 483 500 501 512 513 524 525 536 537 548 558 559 560

```
            562   563   564   565   568   569   578   579   580   581   583   584   585   587   589
            591   594   596   597   598   609   612   613   616   617   635   644   653   654   655
            660   676   677   679   684   704   705   706   713   714   715   722   723   724   738
            739   740   745   752   753   754   759   760   767   784   809   810   811   812   813
            815   817   818   819   822   823   824   830   831   832   836   837   841   842   846
            847   860   869   872   873   874   875   876   877   878   879   881   882   883   885
            886   902   907   909   910   911   912   913   914   915   916   919   920   921   931
            932   933   936   948   952   960   962   965   968   971   973   975   978   979   980
            983   997  1006  1007  1008  1009  1013  1014  1015  1017  1018  1019  1020  1027  1037
           1043  1048  1052  1060  1061  1062  1068  1070  1071  1072  1078  1079  1080  1086  1090
           1094  1098  1106  1107  1109  1110  1114  1115  1117  1118  1121  1127  1128  1132  1140
           1173  1174  1175  1185  1212  3261
MOVB         98   117   130   601   603   627   628   629   630   636   645   870   922  1047  1201
           1203  1205  1207  3261
NOP         981   982   997  3268  3269
RESET        68   135   987   997  3263
ROR        1209  1210  1211
RTI          69   144   146   148   949   954  1050  3261  3267
RTS         115   330   468   610   620   631   640   648   763   827  1065  1075  1193  1208  1215
SUB         125   149   391   399   747  1133  1141
TST         137   505   517   529   541   671   695   710   719   728   774   797   939   946  1038
           1100  1102  3261  3264
TSTB         90   944  1187
WAIT       1177
.ASCII       24  1268  1270  1272  1274  1276  1278  1280  1282  1284  1286  1288  1290  1292  1294
           1296  1298  1300  1302  1304  1306  1308  1310  1312  1314  1316  1318  1326  1333  2024
           2025  2026  2046  2057  2068  2079  2677  2680  2721  2722  2724  2725  2741  2742  2744
           2745  2759  2763  2766  2769  2772  2828  2847  2868  2873  2878  2885  2898  2911  2924
           2937  2950  2963  2976  3013  3014  3015  3020  3168  3171  3174  3177  3234  3237  3240
           3243  3246  3249  3252
.ASCIZ       24  1320  2034
.BYTE        24  1269  1271  1273  1275  1277  1279  1291  1283  1285  1287  1289  1291  1293  1295
           1297  1299  1301  1303  1305  1307  1309  1311  1313  1315  1317  1319  1988  1990  1992
           1994  1995  2027  2056  2067  2078  2541  2554  2567  2575  2613  2624  2678  2681  2695
           2696  2698  2699  2723  2743  3012  3016  3017  3021
.ENABL        4
.END       3307
.ENDC        12    13    18    19    21    23    24    27    69   270   277   284   322   335   344
            356   432   475   480   497   547   557   576   699   732   737   758   765   781   808
            859   868   906   997  1005  3261
.EQUIV       13
.EVEN        24
.IF          12    13    18    19    21    23    24    27    69   270   277   284   322   335   344
            356   432   475   480   497   547   557   576   699   732   737   758   765   781   808
            859   868   906   997  1005  3261
.IFF         13    18    21    23    24    69   270   277   284   322   335   344   356   432   475
            480   497   547   557   576   699   732   737   758   765   781   808   859   868   906
            997  1005  3261
.IFT       3261
.IFTF      3261
.IIF         12    18    19    24    69   997  3261
.IRP         27   270   277   284   335   344   356   432   475   480   497   547   557   576   699
            732   737   765   781   808   859   868   906  1005  3261
.LIST         2    11    13    18    19    24    27    69   270   277   284   335   344   356   432
```

```
          475    480    497    547    557    576    699    732    737    765    781    808    859    868    906
          997   1003   1005   1057   1261   2099   2660   2861   3261
.MACR      34
.MACRO     18     24     69    262   1232   1692   2214   2672   2687   2777   2803   3255
.MCALL      7      8      9     10     13     24     69
.MEXIT     24
.NLIST      1      3     13     19     19     24     27     69    270    277    284    335    344    356    432
          475    480    497    547    557    576    699    732    737    765    781    808    859    868    906
          997    999   1005   1055   1252   2092   2655   2861   3261
.PAGE      24
.REPT      19     24   1620   2135   2152   2168   2336   2346   2356   2367   2677   2680   2695   2698   2850
.SBTTL     13     18     19     21     23 2[illegible]   29     38     66     69    116    153    265    266    267
          268    270    277    284    335    344    356    432    475    480    497    547    557    576    699
          732    737    765    781    808    859    868    906    995    997   1000   1001   1002   1005   1056
         1059   1067   1077   1167   1198   1221   1253   1254   1255   1256   1257   1258   1259   1260   1339
         1491   1616   1632   1685   1686   1933   1951   1986   2000   2038   2102   2127   2145   2162   2225
         2242   2283   2296   2316   2391   2413   2465   2527   2639   2703   2704   2705   2708   2715   2733
         2752   2776   2822   2841   2862   2986   3004   3005   3025   3026   3027   3261
.TITLE     12
.WORD      19     21     23     24    997   1223   1224   1225   1226   1227   1228   1229   1230   1719   1720
         1721   1722   1723   1724   1725   1726   1727   1728   1729   1730   1731   1732   1733   1734   1735
         1736   1737   1738   1739   1740   1741   1742   1743   1744   1745   1746   1747   1748   1749   1750
         1751   1752   1753   1754   1755   1756   1757   1758   1759   1760   1761   1762   1763   1775   1776
         1777   1778   1779   1780   1781   1782   1783   1784   1785   1786   1787   1788   1789   1790   1791
         1792   1793   1794   1795   1796   1797   1798   1799   1800   1801   1802   1803   1804   1805   1806
         1807   1808   1809   1810   1811   1812   1813   1814   1815   1816   1817   1818   1819   1830   1831
         1832   1833   1834   1835   1836   1837   1838   1839   1840   1841   1842   1843   1844   1845   1846
         1847   1848   1849   1850   1851   1852   1853   1854   1855   1856   1857   1858   1859   1960   1861
         1862   1863   1864   1865   1866   1867   1868   1869   1870   1871   1872   1873   1874   2285   2286
         2287   2288   2289   2290   2291   2292
```

ERRORS DETECTED: 0

```
*DSKZ:DZVSDB,DZVSDB/CRF=DZVSDB
RUN-TIME: 74 38 6 SECONDS
CORE USED:  19K
```