

WRUN, JENNY AX11,01620COH9101,0ICKEY,1,25

WNOB SLIB:AX11/PLOT(4)

QSYM PRINTR,,NSP

WASG,AX SLIB,,F2
FAC WARNING 040200000000

WASB,AX R77 ,,F2

WASG,AX ARC*FURNACE77 ,,F2

WUSE FILE,,ARC*FURNACE,

WFOR, SX SLIB,AX11/PLOT,R,AX11/PLOT
FOR 01-3-06/06/79-13:57:04 19,1

SUBROUTINE AX11 ENTRY POINT 001155
AX11S ENTRY POINT 001160

STORAGE USED: CODE(1) 001407; DATA(0) 000207; BLANK COMMON(2) 000000

EXTERNAL REFERENCES (BLOCK, NAME)

0003 SCALED
0004 PLOT
0005 NUMBER
0006 SYMBOL
0007 NIDCB9
0010 NNPTS9
0011 NSCR8
0012 NSTOPS
0013 COS
0014 SIN
0015 XPR1
0016 ALOG10
0017 XPI1

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001	000005	10L	0001	000121	100L	0001	001365	100L	0001	001134	1000L	0001	001417	1000L
0001	000155	120L	0000	000121	141D	0001	000426	140L	0001	000430	180L	0001	000015	20L
0001	001400	200L	0001	000450	200L	0001	000364	220G	0001	000417	230G	0001	000452	240L
0001	000436	243G	0001	000670	260L	0001	000570	275G	0001	000050	30L	0001	001410	300L
0001	000700	315G	0001	001006	320L	0000	000075	970F	0000	R 000061	DLBL	0000	R 000021	DS
0000	R 000015	DX	0000	R 000016	DY	0000	000136	DYNS	0000	R 000135	DYNS	0000	R 000134	DYNS
0000	000133	DYNS	0000	000131	DYNS	0000	R 000027	FIRSMX	0000	R 000023	FIRSI	0000	R 000062	FNEAR
0001	R 001205	FNEAR	0000	R 000003	FUZZ	0000	I 000041	I	0000	L 000070	INV	0000	I 000066	J
0000	I 000083	J	0000	I 000044	K	0000	I 000067	K	0000	I 000045	KSPC	0000	I 000006	KVAL
0000	I 000063	LINT	0001	I 001255	LINT	0000	I 000035	NCHRS	0000	I 000036	NCHRW	0000	I 000032	NDIGIT
0000	I 000031	NDS	0001	I 001315	NRINT	0000	I 000065	NRINT	0000	I 000054	NSIDE	0000	I 000046	NSIG
0000	I 000040	NSPC	0000	I 000012	NTEN	0000	I 000025	NTIK	0000	I 000030	NTIKMY	0000	I 000026	NTIKD
0000	L 000004	HULLED	0000	R 000024	ORIGIN	0000	R 000000	PI	0000	R 000014	RAOS	0000	R 000047	SIDE
0000	R 000052	SIDEX	0000	R 000053	SIDEY	0000	R 000050	SIDXX	0000	R 000051	SIDYY	0000	R 000034	SIZ
0000	R 000002	SIZCHR	0000	R 000013	SPC	0000	L 000005	SPCFLG	0000	R 000033	SPCMX	0000	R 000037	SPX
0000	R 000022	SD	0000	R 000060	SOPDS	0000	R 000057	TIK	0000	R 000001	TIKSI2	0000	R 000064	TNP
0000	R 000042	VAL	0000	R 000055	X	0000	R 000017	XI	0000	R 000056	Y	0000	R 000020	YI

00101 1* SUBROUTINE AX11 000000
00101 2* 6 (XO, YO, LABEL, NCHR, DIST, ANGLE, FIRS, DEL) 000000
00101 3* C 000000

SLIB:AX11/PLOT(4)

DATE 060679

PAGE 2

00101	4*	C	XO	* X-ORIGIN (INCHES)	000000
00101	5*	C	YO	* Y-ORIGIN (INCHES)	000000
00101	6*	C	LABEL	* AXIS LABEL (STRING)	000000
00101	7*	C	NCHR	* # OF CHARACTERS IN LABEL	000000
00101	8*	C	DIST	* AXIS LENGTH (INCHES)	000000
00101	9*	C	ANGLE	* ANGLE AT WHICH AXIS IS ORIENTED (DEGREES)	000000
00101	10*	C	FIRS	* FIRST-VALUE UNITS	000000
00101	11*	C	DEL	* SCALING FACTOR (UNITS/INCH)	000000
00101	12*	C			000000
00101	13*	C	SPC	* NOMINAL SPACING BETWEEN NUMBERS (INCHES)	000000
00101	14*	C		* IF 'SPC' IS NEGATIVE, ALLOW ROUTINE TO PLOT	000000

0001	000438	2438	0001	000430	2430	0001	000430	2430	0001	000430	2430
0000	R 000015	315G	0000	R 000015	315G	0000	R 000015	315G	0000	R 000015	315G
0000	R 000133	DYNE	0000	R 000133	DYNE	0000	R 000133	DYNE	0000	R 000133	DYNE
0001	R 001205	FNEAR	0000	R 000003	FUZZ	0000	R 000027	FIRSHX	0000	R 000023	FIRSI
0000	I 000043	J	0000	I 000044	K	0000	I 000041	L	0000	L 000070	INV
0000	I 000063	LINT	0001	I 001255	LINT	0000	I 000067	K	0000	I 000045	K5PC
0000	I 000031	NDS	0001	I 001315	NRINT	0000	I 000035	NCHRS	0000	I 000036	NCHRW
0000	I 000040	N5PC	0000	I 000017	NTEN	0000	I 000045	NRINT	0000	I 000054	NSIOE
0000	L 000004	NULLED	0000	R 000024	ORIGIN	0000	I 000025	NTIK	0000	I 000030	NTIKHY
0000	RR 000052	SIZEFX	0000	R 000053	SIDEX	0000	R 000000	PI	0000	R 000014	RADS
0000	RR 000004	SIZECHR	0000	R 000013	SPC	0000	R 000050	SIOXX	0000	R 000051	SIOYY
0000	R 000022	SD	0000	R 000060	SOPDS	0000	L 000005	SPCPLG	0000	R 000033	SPCHX
0000	R 000042	VAL	0000	R 000055	X	0000	R 000057	TIK	0000	R 000001	TIKSLY
			0000	R 000017	XI	0000	R 000054	Y	0000	R 000020	YI

```

00101 1* SUBROUTINE AX11
00101 2* 6 IX0, YO, LABEL, NCHR, DIST, ANGLE, FIRS, DEL
00101 3* C
000000
000000
000000

```

SLIB:AX11/PLOT(4)

DATE 060679 PAGE 2

```

00101 4* C XO = X*ORIGIN (INCHES)
00101 5* C YO = Y*ORIGIN (INCHES)
00101 6* C LABEL = AXIS LABEL (STRING)
00101 7* C NCHR = # OF CHARACTERS IN LABEL
00101 8* C DIST = AXIS LENGTH (INCHES)
00101 9* C ANGLE = ANGLE AT WHICH AXIS IS ORIENTED (DEGREES)
00101 10* C FIRS = FIRST VALUE (UNITS)
00101 11* C DEL = SCALING FACTOR (UNITS/INCH)
00101 12* C
00101 13* C SPC = NOMINAL SPACING BETWEEN NUMBERS (INCHES)
00101 14* C 02/APR/79: IF 'SPC' IS NEGATIVE, ALLOW ROUTINE TO PLOT
00101 15* C #15 ON END OF AXIS.
00101 16* C
00101 17* C MAX11 IS DESIGNED AS A PARTIAL REPLACEMENT FOR MAXIS,
00101 18* C TO ALLOW AX11 WITH SPACING LESS THAN ONE INCH.
00101 19* C
00101 20* C CYCLE 0 12-AUG-77
00101 21* C CYCLE 1 17-APR-78 CORRECTIONS TO TIK-PLACING
00101 22* C CYCLE 2 14-FEB-79 OFTN CONVERSION
00101 23* C CYCLE 3 29-MAR-79 CORRECTIONS TO 'NSIG', 'NDS'
00101 24* C CYCLE 4 6 JUN 79 MODIFICATION OF TIK-RESOLUTION
00101 25* C
00103 26* C REAL XO, YO, DIST, ANGLE, FIRS, DEL
00104 27* C INTEGER LABEL, NCHR
00104 28* C
00105 29* C REAL PI / 3.141592653589 /, TIKSIZ / .04/, SIZECHR / .14/
00111 30* C REAL FUZZ / .0005 /
00113 31* C LOGICAL NULLED, SPCFLG
00114 32* C PARAMETER MXXVAL = 4
00115 33* C INTEGER KVAL(MXXVAL) / 1, 2, 5, 10 /
00117 34* C INTEGER NTEN / ' * 10' /
00117 35* C NEW
00121 36* C DEFINE LIM(I,J,K) = MAX(J, MIN(I,K))
00121 37* C
00122 38* C 970 FORMAT('OAX11/PLOT: BAD DATA',
00122 39* C /, ' ', T20, 'DIST', T30, 'DEL', T40, 'NCHR', T50, 'SPC',
00122 40* C /, ' ', T18, 'P, 2E10.2, 15, 5X, E10.2)
00122 41* C
00123 42* C SPC = 1.
00124 43* C GO TO 10
00124 44* C
00124 45* C
00125 46* C ENTRY AX11
00125 47* C (IX0, YO, LABEL, NCHR, DIST, ANGLE, FIRS, DEL, SPC) *****
00127 48* C SPC = SPCP
00127 49* C
00130 50* C 10 CONTINUE
00131 51* C SPCFLG = .TRUE.
00132 52* C IF(SPC .GE. 0.) GO TO 20
00134 53* C SPCFLG = .NOT. SPCFLG
00135 54* C SPC = ABS(SPC)
00136 55* C 20 CONTINUE
00137 56* C IF(DIST .GT. 0. .AND. DEL .NE. 0.
00137 57* C .AND. NCHR .NE. 0 .AND. SPC .GT. 0.01) GO TO 30
00141 58* C PRINT 970, DIST, DEL, NCHR, SPC
00147 59* C STOP
00150 60* C 30 CONTINUE
00151 61* C RADS = ANGLE * PI / 180
00152 62* C DX = COS(RADS)
00153 63* C DY = SIN(RADS)
00154 64* C XI = DIST * DX + XO
00155 65* C YI = DIST * DY + YO
00155 66* C
00155 67* C COMPUTE:
00155 68* C CALL SCALED: RE=COMPUTE SCALING CONSTANTS
00155 69* C DS := INTERVAL SIZE (RESOLVED TO MANTISSA 2,3,4,...,10,12,...)
00155 70* C SO := DISTANCE TO FIRST TIK-MARK (MUST BE NON-NEGATIVE !)
00155 71* C
00156 72* C CALL SCALED(FIRS, DEL, DIST, DS, SO)
00156 73* C
00157 74* C FIRSI = FIRS + DEL * DIST B VALUE ON OTHER END OF AXIS
00160 75* C ORIGIN = -FIRS / DEL B DISTANCE (SIGNED) TO AXIS ORG
00161 76* C OS = (10. ** LINT(OS*DEL)) / DEL B LIMIT TO EFFECTIVE POWER
00161 77* C
00162 78* C 100 CONTINUE
00163 79* C IF(SO .GE. DS) SO = AMOD(SO, DS)
000050
000050
000053
000056
000061
000065
000065
000065
000065
000065
000070
000070
000076
000102
000105
000105
000121
000121

```



```

00165 80* IF(HEAR(SO/DS, 1) .LT. FUZZ) SO = 0.
00167 81* IF(SPC .GE. DS) GO TO 120
00171 82* DS = DS / 10
00172 83* GO TO 100
00172 84* C
00172 85* COMPUTE:
00172 86* NTIK := NUMBER OF TIK'S, STARTING AT 'SO',
00172 87* COUNT TIKS FROM 0 TO NTIK.
00172 88* NTIKD := 'ORIGIN' OF TIK-MARKS (USED FOR PLACEMENT OF TIK-MARKS
00172 89* SO THAT ORIGIN IS TRAPPED ON A LARGE TIK-MARK.)
00172 90* NOTE THAT NTIKD MAY NOT BE IN RANGE (0,NTIK), IN WHICH
00172 91* CASE, NO ORIGIN IS TRAPPED.
00172 92* C
00173 93* 120 CONTINUE
00174 94* NTIK = Nrint((DIST = SO) / DS)
00175 95* NTIKD = Nrint((ORIGIN = SO) / DS)
00175 96* C
00175 97* COMPUTE:
00175 98* NDS := EXPONENT OF 'DS'-SIZED SCALE.
00175 99* NSIG := # OF DIGITS TO PLOT AS FRACTION IN ANNOTATION
00175 100* NDIGIT := APPROX # OF SPACES TO ALLOW FOR AXIS #'S
00175 101* NULLED := FLAG SET IFF AXIS #'S ARE CLOSE ENOUGH TO UNITY THAT
00175 102* NO POWER-OF-TEN ADJUST IS NECESSARY.
00175 103* NCHRW := NUMBER OF CHARACTERS' WIDTH (AMOUNT OF SPACE FOR LABEL)
00175 104* NSPC := NUMBER OF SPACES PER LARGE TIK-MARK. (AXIS IS NUMBERED
00175 105* ONLY ON LARGE TIK-MARKS.)
00175 106* SPX := DISTANCE BETWEEN LARGE TIK-MARKS.
00175 107* SIZ := CHARACTER SIZE FOR ANNOTATION ( #'S AND LABEL )
00175 108* ASSUME SPACE FOR AXIS #'S IN FORM '*NN.MH'
00175 109* C
00176 110* FIRSMX = MAX(ABS(FIRS), ABS(FIRS)) @ MAX VALUE ON AXIS
00177 111* NTIKMX = MAX(NTIK, LIM(NTIK, 0, NTIK),
00177 112* LIM(NTIK, 0, NTIK))
00177 113* C
00200 114* NDS = LINT(DS * DEL) @ EST: SCALING FACTOR
00201 115* NDIGIT = LINT(FIRSMX) - NDS + 1 + 4 @ EST: # OF DIGITS
00201 116* C
00202 117* SPCMX = (DIST * NTIKMX) / MAX(1, NTIK) @ LONGER SECTION LENGTH
00203 118* SIZ = MIN(SIZCHR, SPCMX/NDIGIT) @ AT LEAST 2 #'S/LINE
00204 119* NULLED = .FALSE.
00205 120* IF( ABS(LINT(FIRSMX)) .LE. NDIGIT-4 .AND. NDIGIT .GE. 5)
00205 121* .OR. NDS .EQ. 0)
00205 122* NULLED = .TRUE.
00207 123* IF(NULLED) NDS = 0
00207 124* C
00211 125* NCHRS = ABS(NCHR)
00212 126* NCHRW = NCHRS
00213 127* IF(.NOT. NULLED) NCHRW = NCHRW + 4 @ SIZE OF SCALE FACTOR
00215 128* SPX = MIN(SPCMX, NDIGIT*SIZ)
00216 129* NSPC = MIN(NTIKMX, MAX(1, IFIX(SPX/DS + 1.-FUZZ) ) )
00216 130* C
00216 131* 'NSPC' ESTIMATE ABOVE IS A LOWER-BOUND. ADDN'L CONSTRAINT
00216 132* BY RESTRICTING THE MANTISSA OF NUMERIC LABELS:
00216 133* J := INTEGER PART OF MANTISSA (1,2,3,...,10)
00216 134* IF NO-RANGE FOR IMPROVEMENT, MAY DEFAULT TO 'NTIKMX' IFF
00216 135* AT LEAST 2 #'S WILL BE PRINTED ON AXIS
00216 136* C
00217 137* DO 180 I = NSPC, NTIKMX
00222 138* VAL = 1 @ NOTE: (1.GT.0)
00223 139* VAL = VAL / 10. ** LINT(VAL)
00224 140* J = VAL
00225 141* IF(J .NE. VAL) GO TO 180
00227 142* DO 160 K = 1, MXKVAL
00232 143* IF(KVAL(K) .NE. J) GO TO 160
00234 144* NSPC = I
00235 145* GO TO 240
00236 146* 160 CONTINUE
00240 147* 180 CONTINUE
00240 148* C
00242 149* DO 200 I = NSPC, NTIKMX
00245 150* CALL DIVISE(I, K)
00246 151* IF(K .EQ. 1) GO TO 200
00250 152* NSPC = I
00251 153* GO TO 240
00252 154* 200 CONTINUE
00252 155* C

```

```

00254 156* 240 CONTINUE
00255 157* CALL DIVISE(NSPC, KSPC)
00256 158* IF(.NOT. NULLED)
00256 159* NDS = LINT(ABS(FIRS)+ABS(FIRS))/2 )
00260 160* NSIG = NDS - LINT(DS*NSPC*DEL)
00261 161* IF(NSPC .GT. 10 .AND. NSPC .LT. 20) NSIG = NSIG + 1
00261 162* C
00261 163* COMPUTE SIDE-VECTOR (NOTE THAT ASSUMED DIRECTION FOR
00261 164* TIK'S IS SAME AS SIDE<X,Y>.)
00261 165* C
00261 166* C
00263 167* SIDE = PI/2
00264 168* SIOXX = COS(RADS + SIDE)
00265 169* SIOYY = SIN(RADS + SIDE)
00266 170* IF(INCHR .LT. 0) SIDE = -SIDE
00270 171* SIDEX = COS(RADS + SIDE)
00271 172* SIOY = SIN(RADS + SIDE)
00272 173* NSIDE = - ( INCHR/ABS(INCHR) ) - 1 / 2 @ 0 IF +, 1 IF -
00272 174* C
00272 175* DRAW AXIS LINE, MARKING IT WITH TIK'S
00272 176* C
00273 177* CALL PLOT(X0, Y0, 3)
00273 178* DO 240 I = 0, NTIK

```


SLIB:AXI1/PLOT14

DATE 040679

PAGE 4

```
240 CONTINUE
CALL DIVISE(NSPC, KSPC)
IF (NOT .NULLD)
  NSIG = LINT(LARSIFINSI*ABSIF:RS11/2)
  IF (NSPC .GT. 10 .AND. NSPC .LT. 20) NSIG = NSPC * 1
  NSIG = NSPC * 1

COMPUTE SIDE=VECTOR INOTE THAT ASSUMED DIRECTION FOR
TICK'S IS SAME AS SIDE(X,Y).

SIDE = R1/2
SIDEY = COSIRADS * SIDE
SIDEY = SINIRADS * SIDE
INCHR = LT, 0) SIDE = - SIDE
SIDEY = COSIRADS * SIDE
SIDEY = SINIRADS * SIDE
NSIDE = 1 INCHR/ABSINCHR) - 1) / 2      R 0 IF +, 1 IF -

DRAW AXIS LINE, MARKING IT WITH TICKS

CALL PLOTIXO, Y0, 3)
DO 260 I = 0, NTIK
  IF (MOD(ABS(I-NTIK), NSPC) .NE. 0) GO TO 240
  X = X0 + DX * (50 + DS*I)
  Y = Y0 + DY * (50 + DS*I)
  CALL PLOTIX, Y, 2)
  TIK = TIKSIZ
  IF (NSPC .NE. KSPC .AND. MOD(ABS(I-NTIK), NSPC) .EQ. 0)
    TIK = 2 * TIK
  CALL PLOTIX + TIK * SIDEY, Y + TIK * SIDEY, 2)
  CALL PLOTIX, Y, 2)
240 CONTINUE
CALL PLOTIX1, Y1, 2)

GO BACK & PUT IN NUMBERS

DO 320 I = 0, NTIK
  IF (MOD(ABS(I-NTIK), NSPC) .NE. 0) GO TO 320
  SOPS = 50 + DS*I
  IF (ISPCFLG .AND. (SOPS + NDIGIT*SIZ .GT. DIST)) GO TO 320
  VAL = (FIRS + SOPS * DEL) / (10. ** NDS)
  X = X0 + IDX * (SOPS * SIZ/5) + (SIDEY * SIZCHR*(1+NSIDE))
  Y = Y0 + IDY * (SOPS * SIZ/5) + (SIDEY * SIZCHR*(1+NSIDE))
  CALL NUMBER(X, Y, SIZ, VAL, ANGLE, NSIG)
320 CONTINUE

PUT LABEL ON PLOT-AXIS, WITH POWER-OF-TEN SCALE
IF SUFFICIENT ROOM ON AXIS, CENTER THE LABEL, ELSE RESTRICT TO
START ONE CHARACTER POSITION PAST THE AXIS BEGINNING.

DLBL = MAX(SIZ, (DIST-NCHR*SIZ)/2)
X = Y0 + SIDEY * (SIZ*(3+NSIDE)) + DX * DLBL
Y = Y0 + SIDEY * (SIZ*(3+NSIDE)) + DY * DLBL
CALL SYMBOL(X, Y, SIZ, LABEL, ANGLE, NCHR5)

=> PUT IN EXPONENT OF SCALE FACTOR:

IF (INDS .EQ. 0) GO TO 1000
CALL SYMBOL(999., 999., SIZ, NTEN, ANGLE, 5)
IF (INDS .EQ. 1) GO TO 1000
X = X + (NCHR-1) * SIZ * DX + SIDXX * SIZ*.75
Y = Y + (NCHR-1) * SIZ * DY + SIDYY * SIZ*.75
CALL NUMBER(X, Y, SIZ*.7, FLOAT(INDS), ANGLE, -1)

1000 CONTINUE
RETURN

REAL FUNCTION FNEAR(A,B)
RETURNS DISTANCE TO NEAREST MULTIPLE OF 'A' FROM 'A'

REAL A,B
FNEAR = MIN(ABS(MOD(A,B)), ABS(B - MOD(A,B)))
RETURN
```

000417
000422
000424
000431
000431
000431
000431
000434
000441
000444
000444
000452
000452
000452
000454
000454
000456
000500
000500
000500
000500
000500
000500
000521
000524
000531
000534
000541
000547
000552
000552
000552
000552
000561
000570
000570
000577
000607
000617
000623
000625
000625
000647
000647
000663
000671
000671
000671
000671
000671
000700
000700
000707
000715
000732
000742
000760
000776
001007
001007
001007
001007
001007
001022
001036
001045
001045
001045
001045
001054
001054
001065
001070
001104
001114
001114
001134
001134
001163
001163
001163
001163
001163
001163
001200
001213


```

INTERER FUNCTION LINT1(X)
  RETURNS *BASE* POWER OF TEN, ACCOUNTING FOR ROUND OFF
  REAL TMP
  TMP = ALOSIO(ABS(X))
  IF (TMP >= 0.1) LINT = TMP * FUZZ
  IF (TMP < 0.1) LINT = TMP * 11. * FUZZ
  RETURN

INTERER FUNCTION NRINT1(X)
  RETURNS APPROX INTEGER EQUIV, ACCOUNTS FOR ROUND OFF
  IF (X >= 0) NRINT = X * FUZZ
  IF (X < 0) NRINT = X * FUZZ
  RETURN

SUBROUTINE DIVISE1(SPC, JSPC)
  GIVEN ESTIMATE OF NUMBER SPACING *ISPC*, FIND THE LARGEST
  DIVISOR *JSPC* WITH THE FOLLOWING CONSTRAINTS
    1. JSPC = 1,2,5,10,...
    2. JSPC * DS <= .5 * SPC
  ON EXIT, JSPC = 1,2,5,10,20,...
  INTEGER ISPC, JSPC
  INTEGER J,K
  PARAMETER *KXINV* = 5
  LOGICAL INVINXINV / .TRUE., .TRUE., .FALSE., .FALSE., .TRUE. /
  JSPC = MIN1(SPC, IFIX(SPC/DS * 1. * FUZZ))
  IF (JSPC <= 2) GO TO 1000
  DO 100 J = 5, JSPC, 5
    K = 10 * LINT(FLOAT(JSPC))
    K = MIN(KSPC/K, KXINV)
    CONTINUE
    IF (MOD(INVINXINV, J)) GO TO 200
    JSPC = K * J
    IF (MOD(SPC, JSPC) <= 0) GO TO 300
    CONTINUE
    IF (J <= 1) GO TO 300
    GO TO 100
  200 CONTINUE
    J = J - 1
    GO TO 100
  300 CONTINUE
    J = KXINV
    K = K / 10
    GO TO 100
  1000 CONTINUE
  RETURN
END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

*PACK, P
FURPUR 27R3A (35 SL73R1 06/06/79 13:57:07
END *PACK, TEXT=18, TOC=2, REL=37

```

```

GNAP, SK FILE, PLOTTER, PLOTTER1
MAP 29R1 SL74R1 06/06/79 13:57:09 19)
  1. IN PLOT=RAIN
  2. IN R.TAG/PLOT, SCALER/PLOT, AX11/PLOT, AXUNIT/PLOT, DASHED/PLOT, BOX/PLOT
  3. IN R.GETCHR=FUNC/FLDATA, PUTCHR/FLDATA
  4. IN R.TMARK/PLOT, NEWPEN/PLOT
  5. IN R.SEGMNT/PLOT, PLTKEY/PLOT
  6. IN R.FPNUM/PLOT, SYMBOL/CALCOMP, NUMBER/PLOT
  7. LIB FILE, F/PLOT, F1/PLOT, ...
  8. LIB LIBRARY=FORTRAN, LIBRARY=R, MONITOR=FOR,
  9. LIB LIBRARY=IMSL

```

```

10. END
AFCH STATUS OF OUTPUT ELEMENT=UNKNOWN

ADDRESS LIMITS 001000 033131 13402 18ANK WORDS DECIMAL
                040000 125563 27508 08ANK WORDS DECIMAL
STARTING ADDRESS 020166

```

```

SEGMENT RMAINE 001000 033131 040000 125563
F2COCOR 8(2) 040000 040045 19 DEC 74 14:35:02
F2RTHNF 8(2) 040046 040097 16 AUG 77 16:22:24
F2AN6S 8(2) 040050 040057 16 AUG 77 16:16:39
ATANS/FTNINT 8(1) 001000 001041 23 DEC 74 09:10:30
F2ACTIVS/FORFTH 8(1) 001042 001055 12 JAN 78 11:01:29
8(2) 001056 001071
8(5) 001072 001072

CBEPHATHS/SYS71 8(0) 040063 040424 04 NOV 74 21:01:50
SYS71H 24 FEB 75 09:03:00
CBEPFORVS/SYS73 24 SEP 76 21:26:10
ERUS/SYS74R1=WR1 05 MAR 79 15:09:06
OSCR1S/FTNINT 16 AUG 77 16:11:43
CERUS/SYS74R1=WR1 05 MAR 79 15:08:39
FTAUTO 19 SEP 77 19:54:02
8(1) 001110 001645 8(0) 040425 040531
8(3) INFO=010=LC 8(2) BLANK&COMMON
8(1) 001646 001723 8(0) 040532 040624
8(1) INFO=010=LC 8(2) BLANK&COMMON
8(1) 001724 004211 8(0) 040625 041094
8(1) INFO=010=LC 8(2) BLANK&COMMON
8(1) 004212 004233 21 AUG 74 14:46:14

```

TEST MULTI-LOAD SIMULATION

SHORTED <2>

TE014 15:05:02 21 MAR 77
10PT 1

V_{T1} L-N VOLTAGES AT THE FURNACE BUS
V_{T2} (FURNACE TRANSFORMER PRIMARY)
V_{T3}

I₁ COMPENSATED LINE CURRENTS
I₂
I₃

I_{F1} FURNACE CURRENTS (FURNACE
TRANSFORMER PRIMARY)
I_{F2}
I_{F3}

I_{LC1} COMPENSATOR REACTOR (LINE)
I_{LC2} CURRENTS
I_{LC3}

ΔV_{T12} PER CENT L-L VOLTAGE VARIATION AT
THE FURNACE BUS
ΔV_{T23}
ΔV_{T31}

ΔV_{S12} PER CENT L-L VOLTAGE VARIATION AT
THE FLICKER BUS
ΔV_{S23}
ΔV_{S31}

