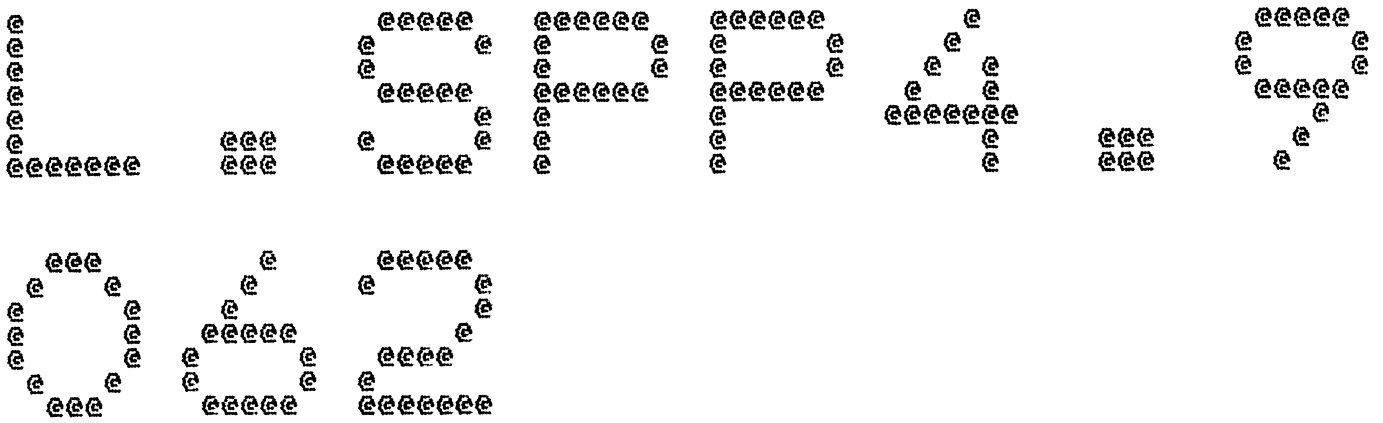
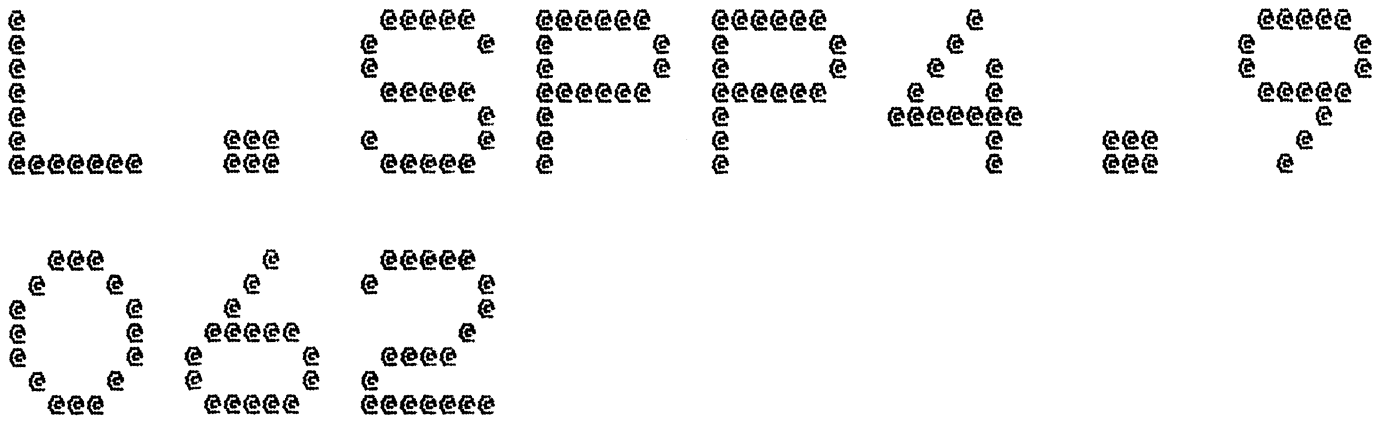


Spool Queue Line #: 1
IRIS LU/Filename : 1/L.SPP4.9062

Printed on/at : MAY 2, 1989 15:21:41
For Group/User: 2 , 2
On Port No: 36

Print control parameters :
Printer Class code : 0
Form Code/paper type : A
Print Priority (0-9) : 5
Starting Page Number : 1
This is copy number : 1
Keep file (Y/N) : N
Notify User when done: N
Comments, optional : For RDC



Spool Queue Line #: 1
IRIS LU/Filename : 1/L.SPP4.9062

Printed on/at : MAY 2, 1989 15:21:49
For Group/User: 2 , 2
On Port No: 36

Print control parameters :
Printer Class code : 0
Form Code/paper type : A
Print Priority (0-9) : 5
Starting Page Number : 1
This is copy number : 1
Keep file (Y/N) : N
Notify User when done: N
Comments, optional : For RDC

```
.EOT   ; JCL FOR SPP4
.EOT
.EOT   ; "BASIC/RUN" R9.4 DEFINITIONS
.EOT   ; "PROTECT" R8.1 SOURCE #1
.EOT   ; "PROTECT" R8.0 SOURCE #2
.EOT   ; "PROTECT" R9.0 SOURCE #3
.EOT   ; "PROTECT" R9.0 SOURCE #4
       .EOT    ; "PROTECT" R9.4 SOURCE #6
       .EOT    ;  'SPP4' R9.2 SOURCE #8
.EOT   ; 'PROTECT' R8.2 SOURCE #7
       .EOT    ;R9XBASSKP
.END
```

```
ASM 1/A.SPP4.9062!,@1/L.SPP4.9062!,B360,-B361,B362,B363
MAY  2, 1989  15:12:02


;       Batchfile:   R94JCL.SPP4

;       ;D=9062  (YDDD)
;       ; NAME = SPP4
;       ; TYPE = 33401
;
;       -R94DEFSPZ
;       -R94BRDEFSE
;       R94PROTECTSC
;       R94SPP4S6A
;       R94SPP4S7B
;       R94BASSKPSB
;       *.END
;
                    .EOT   ; JCL FOR SPP4
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>

; "PROTECT" == "BUSINESS BASIC" PROTECTOR FOR "IRIS" R9.0
; WRITTEN BY  BILL MORITA
; LAST EDIT 11 JUN 83 (3162) by RMS.  (FOR R8.2)
; LAST EDITED FOR R9.0 BY MLR (4275)
; Last edit JPMH - for R9 changes
; Last edit 18 Sep 88 by TWM for Ext BASIC prototype
; Last edit  4 DEC 88 by RDC to remove DSP BREAKPOINTS

;                  All Rights Reserved
;     Copyright (c) 1978, Educational Data Systems
;     Copyright (c) 1979, Educational Data Systems
;     Copyright (c) 1981, POINT 4 Data Corporation
;     Copyright (C) 1983, POINT 4 Data Corporation
;     Copyright (C) 1984, POINT 4 Data Corporation
;     Copyright (C) 1985, POINT 4 Data Corporation
;     Copyright (C) 1986, POINT 4 Data Corporation
;     Copyright (C) 1987, POINT 4 Data Corporation
;     Copyright (C) 1988, POINT 4 Data Corporation
;     Copyright (C) 1989, POINT 4 Data Corporation
;    This document contains secret and confidential
;   information of POINT 4 Data Corporation, and may
;  not be reproduced, used,  or disclosed without the
; prior written permission of POINT 4 Data Corporation


               1          .TXTM  1
             200          .LOC   INFO-400


     200       0 D.SAVE:        0           ;SAVE HEADER BLOCK ADDRESS


     201   32605              DFD     ;POINTER TO DEFAULT DISPLACEMENTS


     202    6330 ERRSO: NCODERROR          ;STORAGE OVERFLOW
     203  101410              3*K!NOP
     204    6330 ERRLN: NCODERROR          ;NO SUCH LINE #
     205  103010              6*K!NOP


     206   41200 .EBS:  EBS
     207   41300 .ESS:  EBS+100
     210   41340 .ESSE: EBS+140
```

```
;                       << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
      211        22 C22:    22          ;CONSTANTS
      212        32 C32:    32
      213        60 C60:    60
      214        71 C71:    71
      215       101 C101:   101
      216       105 C105:   105
      217       112 C112:   112
      220       114 C114:   114
      221       115 C115:   115
      222       121 C121:   121
      223       125 C125:   125
      224       132 C132:   132
      225       136 C136:   136
      226       174 C174:   174
      227       212 C212:   212         ; LF
      230       232 C232:   232         ; ^Z
      231       242 C242:   242         ; "
      232       245 C245:   245         ; %
      233       247 C247:   247         ; '
      234       250 C250:   250         ; (
      235       251 C251:   251         ; )
      236       254 C254:   254         ; ,
      237       256 C256:   256         ; .
      240       272 C272:   272         ; :
      241       273 C273:   273         ; ;
      242       275 C275:   275         ; =
      243       276 C276:   276         ; >
      244       331 C331:   331         ; Y
      245       333 C333:   333         ; [
      246       335 C335:   335         ; ]
      247       336 C336:   336         ; ^

      250       340 C340:   340
      251       342 C342:   342
      252       343 C343:   343
      253       344 C344:   344
      254       350 C350:   350
      255       351 C351:   351
      256       352 C352:   352
      257       353 C353:   353
      260       357 C357:   357
      261       361 C361:   361
      262       362 C362:   362
      263       364 C364:   364
      264       365 C365:   365
      265       366 C366:   366
      266       370 C370:   370
      267       374 C374:   374
      270       375 C375:   375
      271       376 C376:   376

      272    106400 CR:     215*K
      273    157777 MFLAG:  157777
```

```
;                       << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
      274   33127 .ACNL:ACNL    ;POINTERS
      275   32722 .CCAO:CCAO
      276   33133 .COMB:COMB
      277   32730 .CPYP:COPYP
      300   35651 .CRTL:CRTL
      301   37041 .DCOD:DCOD
      302   37666 .DSQ: DSQ     ;DECODE SINGLE QUOTE
      303   34336 .EDEX:EDEX
      304   35176 .EDML:EDML
      305   33261 .EDTR:EDITR
      306   33500 .EDUN:EDUN
      307   34640 .EDXE:EDEXE
      310   35063 .EDXQ:EDEXQ
      311   34515 .EDXS:EDEXS
      312   35057 .EDXX:EDEXX
      313   37640 .ESQ: ESQ     ;ENCODE SINGLE QUOTE
      314   37152 .LSTX:LISTX
      315   35664 .RNUM:RNUM
      316   32517 .NEW: NEW
        317 .PERM = .                   ;DECOY
      317   35631 N9999:EDML9   ;9999 PACKED DECIMAL +1 ; REAL VALUE
;                       ;INSERTED BY SWAPIN
      320   33512 .SENT:SENT
      321   32751 .SFSN:SFSN
      322   33143 .SFVN:SFVN
      323   32473 .SNI: SNI
      324   32461 .SNO: SNO
      325   33051 .SWTB:SWTB
      326   36042 .WORD:WORD
      327   34064 .WTBL:WTBL


       6330 NCODE =JSR @.  ;NCODERROR
      330   32612        ERR

       6331 NEXTB =JSR @.  ;NEXTBYTE
      331   32644        ACNB

       6332 SAVEB =JSR @.  ;SAVE BYTE
      332   32713        STEB
```

```
;                       << SI = R94PROTECTSC; BD = 1/A.SPP4.9062! >>

       333       0 TSE:  0;TEMPORARY STORAGE
       334       0 TSE1:  0
       335       0 TSE2:  0
       336       0 TSE3:  0
       337       0 TSE4:  0
       340       0 TSE5:  0
       341       0 TSE6:  0
       342       0 TSE7:  0

       343       0 LBS:   0        ;LAST BYTE STORED
       344       0 ONF:   0        ;"ON" STATEMENT FLAG
       345       0 SSC:   0        ;SUBSCRIPT STACK COUNTER
       346       0 LNO:   0        ;LINE NUMBER
       347       4 NTES:  4        ;# TABLE ENTRIES TO SEARCH
       350       0 .STMT: 0        ;POINTER TO STATEMENT
       351       0 NEXF:  0        ;"NO EXECUTE" FLAG
       352       0 STYPE: 0        ;STATEMENT TYPE
       353       0 EBP:   0        ;EDITED BYTE POINTER
       354     160 RFBA:  FMAP*2   ;FIRST BYTE OF XOB
```

```
;                       << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
      355 177777 .BUS:  -1        ;BEGINNING OF USER STORAGE (SET BY SWAPI)
      356 177777 .EUS:  -1        ;END OF USER STORAGE (SET BY SWAPI, SWAPO)
      357 100000        0@        ;FOR AFSETUP


           17           .LOC      377-.  ;PAGE ZERO OVERLAP CHECK (PATCH SPACE)
          377           .LOC      377


      377  54333 SWPI:  STA       3,TSE     ;START UP AFTER SWAP-IN
      400 126520         SUBZL     1,1
      401   6101         CALL
      402 100006         LOADUSER
      403 101234         MOVZR#    0,0,SZR   ;[NOP]
      404    414         JMP       SATUP     ;FIND "SAVE"


      405  30004 SWPI1: LDA       2,PIB     ;BEGINNING OF PARTITION
      406  35001         LDA       3,SZP.,2  ;SIZE OF PARTITION
      407  31000         LDA       2,PAD.,2
      410 141000         MOV       2,0
      411 163000         ADD       3,0       ;END OF PARTITION
      412  50355         STA       2,.BUS    ;BEGINNING OF USER STORAGE
      413  40356         STA       0,.EUS    ;END OF USER STORAGE
      414  55010         STA       3,EUS.,2  ;SIZE OF USER STORAGE
      415  34333         LDA       3,TSE
      416   2401         JMP       @.R8FX    ;CHECK R8 FORMAT


      417  32226 .R8FX: R8FIX
```

```
;                       << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
      420   30011 SATUP:LDA     2,.HBA    ;CHECK FOR "SAVE"
      421   25177         LDA     1,DHDR,2
      422  102400         SUB     0,0
      423    4404         JSR     STUP1     ;FIND "SAVE"
      424  151701         .TXTF   ;SA
      425  153305 VE
      426       0 ;

      427  165120 STUP1:MOVZL   3,1
      430    6101         CALL
      431       3         FFILE
      432    6142         TRAPFAULT         ;"SAVE" NOT FOUND !?
      433  100010         NOP
      434   25407         LDA     1,7,3
      435   44200         STA     1,D.SAVE  ;DISC ADDRESS OF "SAVE"
      436   30010         LDA     2,.BSA
      437    6135         READBLOCK         ;READ "SAVE" HEADER
      440   25010         LDA     1,TYPE,2
      441   34410         LDA     3,SFTYP
      442  136414         SEQ     1,3
      443    6142         TRAPFAULT         ;WRONG FILE TYPE FOR "SAVE" !?
      444  100010         NOP
      445   34405         LDA     3,X9999   ;FORM 9999 DECIMAL +1
      446  174242         COMOR   3,3,SZC   ;  AND
      447   54317         STA     3,N9999   ;   SAVE FOR LATER USE
      450     735         JMP     SWPI1

      451   33401 SFTYP:33401   ;  "SAVE" FILE TYPE
      452  146312 X9999:146312


              125         .LOC    INFO-.    ;PAGE ZERO CHECK
```

```
;                       << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
          32200          .LOC    BPS         ;ENTRY POINTERS

32200       377          SWPI    ;SWAP-IN
32201     32274          SWPO    ;SWAP-OUT
32202     32345          CVSP    ;ESCAPE
32203     32345          CVSP    ;CONTROL C

32204       401          JMP     .+1         ;NOP FOR BREAK POINT
32205     20035          LDA     0,C15       ;INITIALIZE DECIMAL PACKAGE
32206      6120          DECIMAL
32207     24410          LDA     1,TRP1      ;SETUP FOR TRACE BUSTER
32210    102400          SUB     0,0
32211     34355          LDA     3,.BUS
32212     41452          STA     0,ERBP.,3
32213     30005          LDA     2,RUP
32214     20354          LDA     0,RFBA
32215     41423          STA     0,ROBP.,3
32216     44401          STA     1,TRP1      ;ZAP TRACEING
32217      6102 TRP1:    FLAGCHANGE          ;RESET "SUPPRESS RETURN IN RUN"
32220    100012          RESET+FLW.
32221      1000          1000
32222      6325          JSR     @.SWTB      ;CHANGE NEXT INSTRUCTION
32223      6401          JSR     @.+1        ;THIS INSTRUCTION IS CHANGED TO JSR @.RNUM
32224     35176          EDML

32225     30237 CLMSK:   30237


32226     30355 R8FIX:   LDA     2,.BUS      ;CHECK FOR R8 FORMAT
32227     25000          LDA     1,REV.,2
32230     20441          LDA     0,A94
32231    106415          SNE     0,1         ;R9.4 FORMAT?
32232       404          JMP     R8FX2       ;   YES, CONTINUE
32233     20434          LDA     0,A80
32234    106654          SUBOR#  0,1,SZR     ;R8 FORMAT ?
32235       410          JMP     OTMS        ;   NO
32236     25005 R8FX2:   LDA     1,SLT.,2    ;   YES, CHECK FOR 1 LINE PRROGRAM SIZE
32237     21004          LDA     0,UPS.,2
32240    122400          SUB     1,0
32241     24002          LDA     1,C2
32242    106432          SGR     0,1         ;PROGRAM TOO SMALL TO PROTECT ?
32243       502          JMP     CVSP        ;   YES, CLEAR PROGRAM
32244      1400          JMP     0,3         ;   NO, RETURN

32245      6133 OTMS:    OUTTEXT
32246    106607          .TXTF   ;<215><207>
32247    150322 PR
32250    147707 OG
32251    151301 RA
32252    146640 M
32253    147317 NO
32254    152240 T
32255    151270 R8
32256    127260 .0
32257    120306  F
32260    147722 OR
32261    146701 MA
32262    152241 T!
```

```
32263       0 ;

32264    6141           STOUTPUT
32265    6101           CALL
32266  100000           SCOPE

32267  134260 A80:      .TXTF   ;80
32270       0 ;

32271  134664 A94:      .TXTF   ;94
32272       0 ;
```

```
;                              << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
    32273   20020                20020
    32274   54333 SWPO:  STA     3,TSE      ;WRAP UP FOR SWAP-OUT
    32275   34100        LDA     3,.INFO    ;we will zero the regnant processor cell
    32276  126400        SUB     1,1        ;  so that a new processor will be loaded
    32277   45443        STA     1,CPDA.,3; and ensure initialized temp cells
                         ; in the unlikely event of two users in PROTECT
    32300   30004        LDA     2,PIB      ;SAVE FILE PROTECTION VALUE
    32301   31003        LDA     2,AFP.,2   ;H(ACTIVE FILE)
    32302   21010        LDA     0,TYPE,2
    32303   40437        STA     0,SWPO2
    32304   34355        LDA     3,.BUS
    32305   21407        LDA     0,NVS.,3   ;SET EUS
    32306  100400        NEG     0,0        ;  =NUS-1
    32307  100000        COM     0,0
    32310   41410        STA     0,EUS.,3
    32311  163000        ADD     3,0        ;SET LAST WORD OF PARTITION
    32312  126400        SUB     1,1        ;  DONT EXPAND USER AREA
    32313  152400        SUB     2,2        ;DONT SWAP TEMPS
    32314    6101        CALL
    32315      67        AFSETUP
    32316   34004        LDA     3,PIB      ;A (PARTITION)
    32317   35403        LDA     3,AFP.,3   ;H(ACTIVE FILE)
    32320   30355        LDA     2,.BUS
    32321   21024        LDA     0,FLAG.,2
    32322   24751        LDA     1,SWPO-1
    32323  107414        AND#    0,1,SZR    ;EXECUTE OR LOAD MODE ?
    32324     413        JMP     SWPO1      ;   YES
    32325   21005        LDA     0,SLT.,2   ;   NO
    32326   25002        LDA     1,PLC.,2
    32327  122414        SEQ     1,0        ;ANY PROGRAM ?
    32330     407        JMP     SWPO1      ;   YES
    32331  101400        INC     0,0        ;   MAYBE
    32332  101400        INC     0,0
    32333   25007        LDA     1,NVS.,2
    32334   31051        LDA     2,ERRN.,2
    32335  151015        SNZ     2,2        ;ANY ERROR,  OR
    32336  122404        SUB     1,0,SZR    ;ANY PROGRAM ?
    32337   20403 SWPO1: LDA     0,SWPO2    ;   YES,  MATCH TYPE
    32340   41410        STA     0,TYPE,3
    32341    2333        JMP     @TSE


    32342       0 SWPO2: 0
```

```
;                              << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
    32343   30234                30234
    32344   77477                77477
    32345   34355 CVSP:  LDA     3,.BUS    ;CLEAR VARIABLES & STACK POINTERS
    32346   25405        LDA     1,SLT.,3
    32347   45402        STA     1,PLC.,3  ;PROGRAM COUNTER
    32350   21412        LDA     0,UFS.,3
    32351   41413        STA     0,UFC.,3  ;USER FUNCTION STACK
    32352   21416        LDA     0,GSS.,3
    32353   41417        STA     0,GSC.,3  ;GOSUB STACK
    32354   21414        LDA     0,FNS.,3
    32355   41415        STA     0,FSC.,3  ;FOR-NEXT STACK
    32356   21406        LDA     0,UVS.,3
    32357   41407        STA     0,NVS.,3
    32360   31411        LDA     2,UFT.,3  ;CLEAR TABLES AND STACKS
    32361  173000        ADD     3,2
    32362  167000        ADD     3,1
    32363  102400        SUB     0,0
    32364   41000        STA     0,0,2
    32365  151400        INC     2,2
    32366  146032        SGE     2,1
    32367     775        JMP     .-3
    32370   21424        LDA     0,FLAG.,3;INITIALIZE FLAG WORD
    32371   24752        LDA     1,CVSP-2
    32372  123400        AND     1,0
    32373  101400        INC     0,0
    32374   41424        STA     0,FLAG.,3
    32375   30100        LDA     2,.INFO   ;INITIALIZE STATUS
    32376   21041        LDA     0,TSC.,2
    32377  101343        MOVOS   0,0,SNC
    32400   20410        LDA     0,.+10
    32401   24777        LDA     1,.-1
    32402   31455        LDA     2,PSTS.,3
    32403  133400        AND     1,2
    32404  146414        SEQ     2,1       ;WAS STATUS PRESET ?
    32405     405        JMP     .+5
    32406  124000        COM     1,1       ;   YES
    32407  123400        AND     1,0
    32410  143000        ADD     2,0
    32411     406        JMP     .+6

    32412  131020        MOVZ    1,2       ;   NO
    32413  101200        MOVR    0,0
    32414  107400        AND     0,1
    32415  132415        SNE     1,2       ;DONE INITIALIZING STATUS ?
    32416     775        JMP     .-3       ;   NO
    32417   41455        STA     0,PSTS.,3;  YES
    32420   31401        LDA     2,VDT.,3  ;PREPARE FOR DEALLOCATE LOOP
    32421  173000        ADD     3,2
    32422   25404        LDA     1,UPS.,3
    32423  167000        ADD     3,1
    32424   34720        LDA     3,CVSP-1
```

```
;                              << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
    32425  21000 CVSP1: LDA     0,0,2      ;DEALLOCATE VARIABLE SPACE
    32426 163400         AND     3,0
    32427  41000         STA     0,0,2
    32430 102400         SUB     0,0
    32431  41001         STA     0,1,2
    32432 151400         INC     2,2
    32433 151400         INC     2,2
    32434 146032         SGE     2,1
    32435    770         JMP     CVSP1
    32436  34355         LDA     3,.BUS
    32437  25405         LDA     1,SLT.,3
    32440  31401         LDA     2,VDT.,3
    32441 132415         SNE     1,2        ;SLT EMPTY ?
    32442    414         JMP     CVSP2      ;   YES
    32443 137000         ADD     1,3
    32444  25400         LDA     1,0,3
    32445 125014         SKZ     1,1        ;LINE #0 EXIST ?
    32446    410         JMP     CVSP2      ;   NO
    32447  44346         STA     1,LNO
    32450 102000         ADC     0,0        ;   YES
    32451  40407         STA     0,CCFLG    ;TELL EDML WE'RE DOING CTRL C
    32452 176400         SUB     3,3
    32453   2304         JMP     @.EDML     ;DELETE LINE #0
    32454 102400 CCDON:  SUB     0,0
    32455  40403         STA     0,CCFLG    ;CLEAR FLAG
    32456   6101 CVSP2:  CALL
    32457 100000         SCOPE

    32460      0 CCFLG:  0
```

```
;                          << SI = R94PROTECTSC; BD = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE ********
    32461   20403 SNO:   LDA     0,SNOR    ;SNO & SNOP & SNI ARE DUMMIES
    32462    2402        JMP     @SNOR

    32463   34005 SNOP:  LDA     3,RUP
    32464   15405 SNOR:  DSZ     OBP.,3
    32465    6120        DECIMAL
    32466    2776        JMP     @SNOR
    32467       0 SNI0:  0
    32470       0 SNI3:  0
    32471       0 SNI2:  0
    32472       0 SNI1:  0


    32473   54777 SNI:   STA     3,SNI1
    32474   20025        LDA     0,C5
    32475   40774        STA     0,SNI2
    32476  102400        SUB     0,0
    32477  167356 SNIF:  ADDOS#  3,1,SEZ
    32500     405        JMP     SNIP
    32501   50766        STA     2,SNI0
    32502    6125        INBYTE
    32503  142724        SUBZS   2,0,SZR
    32504    2766        JMP     @SNI1
    32505   40763 SNIP:  STA     0,SNI3
    32506    6125        INBYTE
    32507    2763        JMP     @SNI1
    32510   14761        DSZ     SNI2
    32511  132401        SUB     1,2,SKP
    32512   20756        LDA     0,SNI3
    32513  103120        ADDZL   0,0
    32514  103120        ADDZL   0,0
    32515  143000        ADD     2,0
    32516     761        JMP     SNIF
```

```
;                          << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
    32517  30355 NEW:   LDA     2,.BUS    ;SET UP FOR NEW PROGRAM
    32520  20471         LDA     0,DFD+4
    32521  41005         STA     0,SLT.,2
    32522  41002         STA     0,PLC.,2
    32523  41001         STA     0,VDT.,2
    32524 101400         INC     0,0
    32525 101400         INC     0,0
    32526  41004         STA     0,UPS.,2
    32527  41006         STA     0,UVS.,2
    32530  41007         STA     0,NVS.,2
    32531  24356         LDA     1,.EUS
    32532 106433         SLE     0,1
    32533   6142         TRAPFAULT         ;USER AREA TOO SMALL FOR STACKS !?
    32534 112410         25*K!NOP
    32535  20450         LDA     0,DFD
    32536  41011         STA     0,UFT.,2
    32537  20447         LDA     0,DFD+1
    32540  41012         STA     0,UFS.,2
    32541  41013         STA     0,UFC.,2
    32542  20445         LDA     0,DFD+2
    32543  41014         STA     0,FNS.,2
    32544  41015         STA     0,FSC.,2
    32545  20443         LDA     0,DFD+3
    32546  41016         STA     0,GSS.,2
    32547  41017         STA     0,GSC.,2
    32550 102400         SUB     0,0
    32551  41051         STA     0,ERRN.,2
    32552  41052         STA     0,ERBP.,2
    32553  41053         STA     0,ERLN.,2
    32554  41055         STA     0,PSTS.,2
    32555  41020         STA     0,DWC.,2
    32556  24053         LDA     1,C200
    32557 125400         INC     1,1
    32560  45024         STA     1,FLAG.,2
    32561  25004         LDA     1,UPS.,2  ;CLEAR TABLES AND STACKS
    32562 147000         ADD     2,1
    32563  20402         LDA     0,.+2
    32564 113001         ADD     0,2,SKP
    32565     25         DET.
    32566 102400         SUB     0,0
    32567  41000         STA     0,0,2
    32570 151400         INC     2,2
    32571 146032         SGE     2,1
    32572    775         JMP     .-3
    32573  30005         LDA     2,RUP     ;CLEAR ACCOUNTING INFO
    32574  25010         LDA     1,ACT.,2
    32575  32004         LDA     2,@PIB
    32576  45007         STA     1,ACNT,2
    32577  41000         STA     0,NAME,2
    32600  41017         STA     0,COST,2
    32601  41020         STA     0,CHGS,2
    32602  41021         STA     0,CHGS+1,2
    32603  41010         STA     0,TYPE,2
    32604   1400         JMP     0,3
```

```
;                                 << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062!  >>
    32605     114 DFD:   UFT      ;DEFAULT DISPLACEMENTS
    32606     146        UFS
    32607     177        FNS
    32610     337        GSS
    32611     457        SLT


    32612   21400 ERR:   LDA      0,0,3      ;GET ERROR CODE
    32613   30355        LDA      2,.BUS
    32614   41051        STA      0,ERRN.,2
    32615    6101        CALL
    32616  100011        WONA
    32617   34355        LDA      3,.BUS
    32620   21451        LDA      0,ERRN.,3
    32621    6101        CALL
    32622      12        ERROR
    32623   34355        LDA      3,.BUS
    32624   41451        STA      0,ERRN.,3
    32625   25424        LDA      1,FLAG.,3
    32626   20042        LDA      0,C20
    32627  123415        AND#     1,0,SNR
    32630     407        JMP      ERR1
    32631    6133        OUTTEXT
    32632  120311        .TXTF    ;  I
    32633  147240 N
    32634       0 ;

    32635   24346        LDA      1,LNO
    32636    6324        JSR      @.SNO
    32637  152400 ERR1:  SUB      2,2
    32640    6132        OUTBYTE

    32641    6141 ERR2:  STOUTPUT
    32642    6101        CALL
    32643  100000        SCOPE


    32644   30355 ACNB:  LDA      2,.BUS     ;ACCESS NEXT PROGRAM BYTE
    32645   25003        LDA      1,PBC.,2
    32646   11003        ISZ      PBC.,2
    32647    2144        JMP      @XGETB&377
```

```
;                          << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;***** THIS PAGE IS GARBAGE *****
    32650   4460 SIZE:  JSR     COPYP
    32651   6133        OUTTEXT
    32652 120275        .TXTF   ; =
    32653 120000   ;

    32654  30355        LDA     2,.BUS
    32655  21005        LDA     0,SLT.,2
    32656 115400        INC     0,3
    32657 161400        INC     3,0
    32660 157000        ADD     2,3
    32661  24356        LDA     1,.EUS
    32662 166400        SUB     3,1
    32663  44334        STA     1,TSE1
    32664  25007        LDA     1,NVS.,2
    32665 106400        SUB     0,1
    32666  20032        LDA     0,C12
    32667 152400        SUB     2,2
    32670   6101        CALL
    32671      7        CIA
    32672   6133        OUTTEXT
    32673 120327        .TXTF   ; W
    32674 147722 OR
    32675 142323 DS
    32676 120317  O
    32677 152724 UT
    32700 120317  O
    32701 143240 F
    32702      0 ;

    32703  24334        LDA     1,TSE1
    32704  20032        LDA     0,C12
    32705 152400        SUB     2,2
    32706   6101        CALL
    32707      7        CIA
    32710 152400        SUB     2,2
    32711   6132        OUTBYTE
    32712   2320        JMP     @.SENT


    32713  20064 STEB:  LDA     0,C377
    32714 143400        AND     2,0
    32715  40343        STA     0,LBS
    32716  24353        LDA     1,EBP
    32717  10353        ISZ     EBP
    32720   2134        JMP     @PUTBY&377


    32721      0        0
    32722  54777 CCAO:  STA     3,.-1
    32723   6106        CHANNEL
    32724     27        CLEAR
    32725   6142        TRAPFAULT
    32726   2773        JMP     @CCAO-1
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
    32727       0          0
    32730   54777 COPYP: STA     3,.-1      ;CHECK COPY PROTECTION
    32731   30355         LDA     2,.BUS
    32732   35001         LDA     3,VDT.,2
    32733  157000         ADD     2,3        ;ABS ADDRESS OF VAR DEF TABLE
    32734   21774         LDA     0,-4,3     ;GET HIGHEST STATEMENT #
    32735   24317         LDA     1,N9999
    32736  122032         SGE     1,0        ;THIS FILE PROTECTED ALREADY?
    32737     407         JMP     CPYP2
    32740   21055         LDA     0,PSTS.,2
    32741   24402         LDA     1,.+2
    32742  123401         AND     1,0,SKP
    32743   20410         LDA     0,.+10
    32744  106414         SEQ     0,1        ;PROGRAM STATUS BAD??
    32745    2762         JMP     @COPYP-1   ;   RETURN

    32746    6330 CPYP2: NCODERROR           ;LIST/COPY PROTECTED
    32747  106010         14*K!NOP
    32750    2757         JMP     @COPYP-1


    32751   30355 SFSN:  LDA     2,.BUS     ;SEARCH FOR STATEMENT NUMBER
    32752   25001         LDA     1,VDT.,2
    32753  147000         ADD     2,1
    32754   44334         STA     1,TSE1     ;ABSOLUTE .VDT
    32755   25005         LDA     1,SLT.,2
    32756  133000         ADD     1,2        ;ABSOLUTE .SLT
    32757   25000 SFSN1: LDA     1,0,2
    32760  106415         SNE     0,1
    32761    1401         JMP     1,3        ;SKIP-RETURN IF FOUND
    32762  106432         SGR     0,1
    32763    1400         JMP     0,3
    32764  151400         INC     2,2
    32765  151400         INC     2,2
    32766   24334         LDA     1,TSE1
    32767  146032         SGE     2,1        ;END OF SLT ?
    32770     767         JMP     SFSN1      ;   NO
    32771  152400         SUB     2,2        ;   YES, NOT FOUND
    32772    1400         JMP     0,3
```

```
;                    << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;      ****** THIS PAGE IS GARBAGE ****
32773     144           144
32774   24777 HELP:     LDA     1,.-1       ;PRINT ERROR MESSAGE
32775  167000           ADD     3,1
32776   34355           LDA     3,.BUS
32777   45451           STA     1,ERRN.,3;MESSAGE NUMBER
33000    6101           CALL
33001  100011           WONA
33002   34355           LDA     3,.BUS
33003  102120           ADCZL   0,0         ;CAUSE LEADING RETURN
33004     423           JMP     HELPC


33005  102400           SUB     0,0         ;TYPE ERROR MESSAGE
33006   40333           STA     0,TSE
33007    6125           INBYTE
33010    6127           ISA2DIGIT           ;ERROR NUMBER ENTERED ?
33011     410           JMP     .+10        ;   NO
33012  132400           SUB     1,2         ;   YES, CONVERT TO OCTAL
33013   20333           LDA     0,TSE
33014  105120           MOVZL   0,1
33015  125120           MOVZL   1,1
33016  123120           ADDZL   1,0
33017  143000           ADD     2,0         ;ACCUMULATE ERROR NUMBER
33020     410           JMP     HELPC+1

33021    6133           OUTTEXT             ;OUTPUT COLON AND ERROR MESSAGE
33022  142710           .TXTF   ;EH
33023  137400 ?;

33024   34355           LDA     3,.BUS
33025   24333           LDA     1,TSE
33026  125015           SNZ     1,1         ;ERROR NUMBER ENTERED ?
33027   25451 HELPC:    LDA     1,ERRN.,3;   NO, GET # OF LAST ERROR
33030    6101           CALL
33031      13           MESSAGE
33032  100010           NOP
33033   30355           LDA     2,.BUS
33034    6102           FLAGCHECK           ;CHANNEL INPUT ?
33035   10024           FLAG.+SKIPZ
33036      20           20
33037     406           JMP     .+6         ;   YES
33040   30005           LDA     2,RUP       ;   NO
33041    6102           FLAGCHECK           ;ECHO DISABLED ?
33042   10012           FLW.+SKIPZ
33043       1           1
33044    2320           JMP     @.SENT      ;   NO
33045    6133           OUTTEXT             ;   YES
33046  106400           215*K
33047    6141           STOUTPUT
33050    2320           JMP     @.SENT

               .EOT   ;  "PROTECT" R8.1 SOURCE #1
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>

;  "BUSINESS PROTECT" SOURCE #2 OF 6 TAPES FOR "IRIS" R8.0

; ********** THIS PAGE IS MOSTLY GARBAGE & IS USED ONLY *******
; ********* TO GET TO 'SFVN' WITH A3 SETUP ********************
; ********* MEANINGFUL INSTRUCTIONS ARE COMMENTED WITH <<<<<<<<<<<<

 33051 151014 SWTB:  SKZ     2,2          ;SEARCH WORD TABLE
 33052  50335        STA     2,TSE2
 33053  54336        STA     3,TSE3
 33054  44334        STA     1,TSE1
 33055  44340        STA     1,TSE5
 33056  24327        LDA     1,.WTBL
 33057 123000        ADD     1,0
 33060  40337        STA     0,TSE4
 33061 151014        SKZ     2,2
 33062  10336        ISZ     TSE3         ;<<<<<<<<<<<<<<<<<<<<
 33063   4460        JSR     SFVN         ;<<<<<<<<<<<<<<<<<<<<
 33064   6315        JSR     @.RNUM       ;<<<<<<<<<<<<<<<<<<<<
 33065  24335        LDA     1,TSE2
 33066 147000        ADD     2,1
 33067  44335        STA     1,TSE2       ;FIRST THREE LETTERS
 33070  22337        LDA     0,@TSE4
 33071 122555        SUBOL#  1,0,SNR
 33072    405        JMP     .+5
 33073  10337        ISZ     TSE4
 33074  14334        DSZ     TSE1
 33075    773        JMP     .-5
 33076   2336        JMP     @TSE3
 33077  10336        ISZ     TSE3
 33100 101112        SSP     0,0
 33101    416        JMP     SWTBD
 33102  30337        LDA     2,TSE4
 33103  21114        LDA     0,WTL,2
 33104  40340        STA     0,TSE5
 33105   4422        JSR     ACNL
 33106   4415        JSR     SWTBC
 33107   4424        JSR     COMB
 33110   4413        JSR     SWTBC
 33111  24334        LDA     1,TSE1
 33112 133000        ADD     1,2          ;SECOND THREE LETTERS
 33113  24340        LDA     1,TSE5
 33114 146554        SUBOL#  2,1,SZR
 33115   6330        NCODERROR            ;UNRECOGNIZABLE WORD
 33116 104410        11*K!NOP
 33117  30337 SWTBD: LDA     2,TSE4
 33120  24327        LDA     1,.WTBL
 33121 132400        SUB     1,2
 33122   2336        JMP     @TSE3

 33123  24340 SWTBC: LDA     1,TSE5
 33124 146555        SUBOL#  2,1,SNR
 33125    772        JMP     SWTBD
 33126  50334        STA     2,TSE1
 33127  54341 ACNL:  STA     3,TSE6       ;ACCESS NEXT LETTER
 33130   6125        INBYTE
 33131   6130        ISA2LETTER
 33132   2341        JMP     @TSE6
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>

;************ THIS PAGE IS MOSTLY GARBAGE.  IT IS USED TO **************
;********** GET TO 'RNUM'.  MEANINGFUL INSTRUCTIONS ARE *************
;********** FLAGGED WITH <<<<<<<<<<<<<<<<.

 33133 151120 COMB:  MOVZL   2,2        ;COMBINE TWO LETTERS
 33134 153120        ADDZL   2,2
 33135  24334        LDA     1,TSE1
 33136 125300        MOVS    1,1
 33137 133120        ADDZL   1,2
 33140 151120        MOVZL   2,2
 33141   1400        JMP     0,3


 33142  57477        57477
 33143  56336 SFVN:  STA     3,@TSE3    ;<<<<<<<<<<<<<<<<<<<<
 33144  20776        LDA     0,.-2
 33145  44341        STA     1,TSE6
 33146 107400        AND     0,1
 33147  30225        LDA     2,C136     ;94
 33150  50337        STA     2,TSE4
 33151  34355        LDA     3,.BUS
 33152  31405        LDA     2,SLT.,3   ;<<<<<<<<<<<<<<<<<<<<<<
 33153 173000        ADD     3,2        ;<<<<<<<<<<<<<<<<<<<<
 33154  34355 SFVN1: LDA     3,.BUS     ;SCAN VARIABLE DEFINITION TABLE
 33155  21404        LDA     0,UPS.,3
 33156 163000        ADD     3,0
 33157  20763        LDA     0,SFVN-1   ;   NO
 33160  35000        LDA     3,0,2      ;<<<<<<<<<<<<<<<<<<
 33161 175112        SSP     3,3        ;ALREADY PROTECTED ?  <<<<<<<<<<<<<<<<<<
 33162    412        JMP     SFVN2      ;   YES     <<<<<<<<<<<<<<<<<<
 33163 117400        AND     0,3
 33164 166415        SNE     3,1        ;IS THIS IT ?
 33165  14337        DSZ     TSE4       ;   NO
 33166 151401        INC     2,2,SKP
 33167   6330        NCODERROR          ;TOO MANY VARIABLES
 33170 104010        10*K!NOP
 33171 151400        INC     2,2
 33172  14336        DSZ     TSE3       ;<<<<<<<<<<<<<<<<<<
 33173   2336        JMP     @TSE3      ;<<<<<<<<<<<<<<<<<<

 33174  25407 SFVN2: LDA     1,NVS.,3   ;NAME NOT IN TABLE
 33175 167000        ADD     3,1
 33176  34356        LDA     3,.EUS
 33177 136433        SLE     1,3        ;TWO WORDS AVAILABLE ?
 33200 176000        ADC     3,3        ;   YES
 33201 167000        ADD     3,1
 33202 122032        SGE     1,0        ;ANYTHING TO BE MOVED ?
 33203 111400        INC     0,2        ;   YES
 33204 151400        INC     2,2
 33205   6133        OUTTEXT;ALREADY PROTECTED!  <<<<<<<<<<<<<<<<
 33206 140714        .TXTF   ;AL
 33207 151305 RE
 33210 140704 AD
 33211 154640 Y
 33212 150322 PR
 33213 147724 OT
 33214 142703 EC
```

```
33215 152305 TE
33216 142000 D;

33217   6141        STOUTPUT          ;<<<<<<<<<<<<<<<<
33220   6101        CALL    ;<<<<<<<<<<<<<<<<<<<<
33221 100000        SCOPE   ;<<<<<<<<<<<<<<<<<<<<
33222  34355        LDA     3,.BUS    ;PUT NEW NAME IN TABLE
33223  31404        LDA     2,UPS.,3
33224 173000        ADD     3,2       ;OLD .UPS
33225  24341        LDA     1,TSE6
33226  45000        STA     1,0,2
33227 102400        SUB     0,0
33230  41001        STA     0,1,2
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE ******
    33231  25405          LDA     1,SLT.,3 ;ADJUST SLT & VDT POINTERS
    33232 137000          ADD     1,3
    33233  24002          LDA     1,C2
    33234  21401          LDA     0,1,3    ;GET A "POINTER"
    33235 101014          SKZ     0,0      ;LOCATION ASSIGNED ?
    33236 123000          ADD     1,0      ;   YES, ADJUST
    33237  41401          STA     0,1,3
    33240 175400          INC     3,3
    33241 175400          INC     3,3
    33242 172032          SGE     3,2      ;END OF VDT ?
    33243    771          JMP     .-7      ;   NO
    33244  34355          LDA     3,.BUS   ;   YES
    33245  11404          ISZ     UPS.,3   ;STEP POINTERS
    33246  11404          ISZ     UPS.,3
    33247  11406          ISZ     UVS.,3
    33250  11406          ISZ     UVS.,3
    33251  11407          ISZ     NVS.,3
    33252  11407          ISZ     NVS.,3
    33253  30247 SFVN3:    LDA     2,C336   ;COMPUTE NEW VARIABLE NUMBER
    33254  20337          LDA     0,TSE4
    33255 112400          SUB     0,2
    33256   2340          JMP     @TSE5


    33257  30251          LDA     2,C342   ;<< ENTRY TO COPY COMMENT
    33260   6332          SAVEBYTE
    33261   6126 EDITR:    INSTBYTE         ;"REM" STATEMENT
    33262  24055          LDA     1,C215
    33263 146415          SNE     2,1      ;END OF LINE ?
    33264    403          JMP     .+3
    33265   6332          SAVEBYTE         ;   NO
    33266    773          JMP     EDITR

    33267 152400          SUB     2,2      ;   YES
    33270   6332          SAVEBYTE
    33271   2401          JMP     @.+1
    33272  34630          EDEXD


    33273  34005 EDITL:    LDA     3,RUP    ;ASSUME "LET" STATEMENT
    33274  25404          LDA     1,IBP.,3
    33275  15404          DSZ     IBP.,3
    33276   6124          GETBYTE
    33277  20056          LDA     0,C240
    33300 142415          SNE     2,0
    33301    772          JMP     EDITL
    33302  30402          LDA     2,.+2
    33303   2473          JMP     @.NCD1
    33304    127          127
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;********* THIS PAGE IS GARBAGE *******
    33305     27        27
    33306  20777 SENTE: LDA    0,.-1     ;ERROR IN READ ITEM
    33307 162414        SEQ    3,0
    33310    414        JMP    SNTCX-3
    33311  20003        LDA    0,C3      ;RECORD LOCKED, PAUSE
    33312   6101        CALL
    33313     57        SIGPAUSE
    33314   6117        BUMPUSER
    33315    430        JMP    SENTC+1

    33316  33317        EICB   ;ENTRY ITEM CONTROL BLOCK
    33317 177777 EICB:  -1     ;RECORD NUMBER = SEQUENTIAL
    33320      0        0      ;ITEM NUMBER = ZERO
    33321     11        11     ;ITEM TYPE = STRING
    33322 177777        -1     ;LENGTH (# BYTES)
    33323 177777        -1     ;BYTE ADDRESS

    33324  20212        LDA    0,C32
    33325 162414        SEQ    3,0       ;END OF FILE ?
    33326    407        JMP    SNTRS     ;  NO
    33327 102400 SNTCX: SUB    0,0       ;CHANNEL ENTRY MODE EXIT
    33330   6106        CHANNEL
    33331     26        CLOSE
    33332 175014        SKZ    3,3
;     JMP     ESCR       *** COMMENTED 01 12 82 BY DJZ
    33333   6142        TRAPFAULT        ;CHANNEL #0 IS ILLEGAL !?
    33334 114010        30*K!NOP

    33335  54333 SNTRS: STA    3,TSE     ;RESET CHANNEL I/O FLAG
    33336  30355        LDA    2,.BUS
    33337   6102        FLAGCHANGE
    33340 100024        RESET+FLAG.
    33341     20        20
    33342  34333        LDA    3,TSE     ;ERROR CODE
    33343   2317        JMP    @.PERM
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062!  >>
;******** THIS PAGE IS GARBAGE ********
    33344    6300 SENTC: JSR     @.CRTL     ;SET ENTRY FROM CHANNEL
    33345   34005        LDA     3,RUP
    33346   21402        LDA     0,FBA.,3
    33347  105400        INC     0,1
    33350   44753        STA     1,EICB+4
    33351   25403        LDA     1,LBA.,3
    33352  106400        SUB     0,1
    33353   44747        STA     1,EICB+3
    33354  102400        SUB     0,0
    33355   40040        STA     0,SBA
    33356   30740        LDA     2,EICB-1
    33357    6106        CHANNEL
    33360      33        READITEM
    33361     725        JMP     SENTE
    33362   34005        LDA     3,RUP
    33363   31402        LDA     2,FBA.,3
    33364   51404        STA     2,IBP.,3
    33365   24735        LDA     1,EICB+3
    33366  106033        SLS     0,1        ;DID LINE FIT IN BUFFER ?
    33367    6330        NCODERROR          ;   NO
    33370  145410        113*K!NOP
    33371   30073        LDA     2,ESCF     ;   YES
    33372  101014        SKZ     0,0        ;EMPTY RECORD . . .
    33373  151014        SKZ     2,2        ;OR ESCAPE PRESSED ?
    33374    4733        JSR     SNTCX      ;   YES
    33375     535        JMP     NCOD       ;   NO, ENCODE THE LINE

    33376   33647 .NCD1: NCOD1

    33377   33671        EDITI
    33400   20241 EDITS: LDA     0,C273     ;SPECIAL CHARACTER
    33401  142414        SEQ     2,0        ;SEMI-COLON ?
    33402     404        JMP     .+4
    33403   30402        LDA     2,.+2      ;   YES, "PRINT"
    33404    2772        JMP     @.NCD1
    33405     134        134

    33406   20225        LDA     0,C136     ;   NO
    33407  151014        SKZ     2,2
    33410  142033        SLS     2,0        ;FROM CARD READER ?
    33411   20221        LDA     0,C115     ;   YES
    33412  142015        ADC#    2,0,SNR    ;"IF" STATEMENT ?
    33413    2764        JMP     @EDITS-1   ;   YES
    33414  142414        SEQ     2,0        ;"ON" STATEMENT ?
    33415    2761        JMP     @.NCD1     ;   NO
    33416   54344 EDITO: STA     3,ONF      ;   YES
    33417   20026        LDA     0,C6
    33420   40347        STA     0,NTES
    33421   30221        LDA     2,C115
    33422    2754        JMP     @.NCD1
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;********** THIS PAGE IS GARBAGE *******
     33423 114631          114631
     33424  37513          FSLN
     33425   6323 DLTE:    JSR     @.SNI       ;"DELETE" COMMAND
     33426  34306          LDA     3,.EDUN
     33427  24346          LDA     1,LNO
     33430 101014          SKZ     0,0         ;TOTAL DELETE ?
     33431    403          JMP     .+3         ;   NO
     33432 125015          SNZ     1,1
     33433   2316          JMP     @.NEW       ;   YES
     33434 101015          SNZ     0,0         ;ENDING LINE # GIVEN ?
     33435  20766          LDA     0,DLTE-2    ;   NO, USE 9999
     33436  30355          LDA     2,.BUS
     33437  41076          STA     0,TSU.+3,2
     33440   6102          FLAGCHANGE          ;SET DELETE FLAG
     33441 140024          SET+FLAG.
     33442   2000          2000
     33443  20346 DLTE1:   LDA     0,LNO       ;<< RE-ENTRY FROM EDML
     33444   6760          JSR     @DLTE-1     ;FIND CURRENT OR NEXT LINE
     33445    433          JMP     EDUN        ;   NOT FOUND, DONE
     33446  30355          LDA     2,.BUS
     33447  45074          STA     1,TSU.+1,2
     33450  34073          LDA     3,ESCF
     33451 175014          SKZ     3,3         ;ESCAPE ?
;      JMP    ESCR        ;   YES *** COMMENTED 01 12 82 BY DZJ
     33452   6300          JSR     @.CRTL      ;   NO, CHECK RUN TIME LIMIT
     33453  30355          LDA     2,.BUS
     33454  21074          LDA     0,TSU.+1,2
     33455  40346          STA     0,LNO
     33456  31076          LDA     2,TSU.+3,2
     33457 112433          SLE     0,2         ;ALL DONE ?
     33460    420          JMP     EDUN        ;   YES
     33461 176400          SUB     3,3         ;   NO
     33462   2304          JMP     @.EDML      ;DELETE LINE # IN AO
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE ********
    33463 177777 QFLAG: 177777
    33464  32460 .CCFG: CCFLG
    33465  32454 .CCDN: CCDON

    33466 135777        135777
    33467  36620        OPLD
    33470   6777 LOAD:  JSR     @.-1        ;"LOAD" COMMAND
    33471  30355        LDA     2,.BUS
    33472  25024        LDA     1,FLAG.,2
    33473  20042        LDA     0,C20
    33474 107000        ADD     0,1
    33475  45024        STA     1,FLAG.,2
    33476 102400        SUB     0,0
    33477  41074        STA     0,TSU.+1,2

    33500  22764 EDUN:  LDA     0,@.CCFG    ;ENCODE DONE
    33501 101014        SKZ     0,0         ;CTRL C PENDING ?
    33502   2763        JMP     @.CCDN      ;   YES
    33503  30355        LDA     2,.BUS
    33504  25024        LDA     1,FLAG.,2
    33505  20761        LDA     0,LOAD-2
    33506 107400        AND     0,1         ;CLEAR "RENUMB", "DELETE" FLAGS
    33507  45024        STA     1,FLAG.,2
    33510   6101        CALL
    33511 100011        WONA
    33512  30355 SENT:  LDA     2,.BUS      ;SET ENTRY MODE
    33513  25024        LDA     1,FLAG.,2
    33514  20042        LDA     0,C20
    33515 123414        AND#    1,0,SZR     ;CHANNEL INPUT ?
    33516    626        JMP     SENTC       ;   YES
    33517  30005        LDA     2,RUP       ;   NO
    33520  21012        LDA     0,FLW.,2
    33521 101213        MOVR#   0,0,SNC     ;IS ECHO ENABLED ?
    33522    407        JMP     ENT1        ;   NO
    33523  20740        LDA     0,QFLAG     ;   YES
    33524 101015        SNZ     0,0         ;QUIET ?
    33525    424        JMP     NC1         ;   YES
    33526   6133        OUTTEXT             ;   NO
    33527 106400        215*K
    33530   6141        STOUTPUT
    33531   6140 ENT1:  STINPUT
```

```
;                          << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE ********
    33532  22004 NCOD:  LDA     0,@PIB    ;ENCODE A STATEMENT
    33533  40040         STA     0,SBA
    33534  34355         LDA     3,.BUS
    33535  25405         LDA     1,SLT.,3
    33536  31401         LDA     2,VDT.,3
    33537 132415         SNE     1,2       ;SLT EMPTY ?
    33540    411         JMP     NC1       ;   YES
    33541 137000         ADD     1,3       ;   NO
    33542  25400         LDA     1,0,3
    33543 125014         SKZ     1,1       ;LINE #0 EXIST ?
    33544    405         JMP     NC1       ;   NO
    33545  44716         STA     1,QFLAG   ;   YES, QUIETLY . . .
    33546  44346         STA     1,LNO
    33547 176400         SUB     3,3
    33550   2304         JMP     @.EDML    ; . . . DELETE IT
    33551 102000 NC1:    ADC     0,0
    33552  40711         STA     0,QFLAG   ;CR IS OK AGAIN
    33553  30207         LDA     2,.ESS
    33554  50345         STA     2,SSC
    33555 102520         SUBZL   0,0
    33556  41000         STA     0,0,2
    33557 102400         SUB     0,0
    33560  42206         STA     0,@.EBS
    33561  40344         STA     0,ONF
    33562  40351         STA     0,NEXF
    33563  40352         STA     0,STYPE
    33564  34355         LDA     3,.BUS
    33565  31401         LDA     2,VDT.,3
    33566 173000         ADD     3,2
    33567  41000         STA     0,0,2
    33570  20003         LDA     0,C3
    33571  40347         STA     0,NTES
    33572  24206         LDA     1,.EBS
    33573  44350         STA     1,.STMT
    33574 125120         MOVZL   1,1
    33575  44353         STA     1,EBP
    33576    402         SKIP
    33577   6125         INBYTE
    33600 102400         SUB     0,0
    33601   6126         INSTBYTE
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE ********
    33602  20053          LDA     0,C200
    33603  24227          LDA     1,C212
    33604 142414          SEQ     2,0           ;NULL . . .
    33605 146415          SNE     2,1           ;OR LF ?
    33606    771          JMP     .-7           ;   YES, IGNORE
    33607  20055          LDA     0,C215        ;   NO
    33610 142014          ADC#    2,0,SZR       ;FORM FEED . . .
    33611 132015          ADC#    1,2,SNR       ;OR VERTICAL TAB ?
    33612    774          JMP     .-4           ;   YES, IGNORE
    33613   6323          JSR     @.SNI         ;   NO, GET LINE NUMBER
    33614  40346          STA     0,LNO
    33615  34055          LDA     3,C215
    33616 156405          SUB     2,3,SNR       ;EMPTY LINE ?
    33617   2304          JMP     @.EDML        ;   YES
    33620   6130          ISA2LETTER            ;LETTER ?
    33621   2443          JMP     @.EDTS        ;   NO
    33622  50334          STA     2,TSE1        ;   YES
    33623   6125          INBYTE                ;GET NEXT BYTE
    33624  34005          LDA     3,RUP
    33625  15404          DSZ     IBP.,3
    33626   6130          ISA2LETTER            ;ANOTHER LETTER ?
    33627   2434          JMP     @.EDTL        ;   NO
    33630   6274          JSR     @.ACNL        ;   YES
    33631   6276          JSR     @.COMB        ;COMBINE TWO LETTERS OF WORD
    33632  34436          LDA     3,EDITI-1
    33633  25400          LDA     1,0,3
    33634 132535          SUBZL#  1,2,SNR       ;"IF" ?
    33635    434          JMP     EDITI         ;   YES
    33636  25401          LDA     1,1,3         ;   NO
    33637 132535          SUBZL#  1,2,SNR       ;"ON" ?
    33640   2425          JMP     @.EDTO        ;   YES
    33641  20042          LDA     0,C20         ;   NO
    33642  24043          LDA     1,C37
    33643   6325          JSR     @.SWTB        ;STATEMENT DIRECTIVE WORD ?
    33644    524          JMP     EDITC         ;   NO
    33645  24213 EDITK:   LDA     1,C60         ;   YES
    33646 133000          ADD     1,2           ;CONVERT TO INTERNAL CODE
    33647  52417 NCOD1:   STA     2,@.STYP
    33650   6332          SAVEBYTE              ;STORE DIRECTIVE BYTE
    33651  24223          LDA     1,C125
    33652 106415          SNE     0,1           ;"DATA" STATEMENT ?
    33653    422          JMP     NCODD         ;   YES
    33654 122015          ADC#    1,0,SNR       ;"REM" STATEMENT ?
    33655   2305          JMP     @.EDTR        ;   YES
    33656  24215          LDA     1,C101        ;   NO
    33657 106433          SLE     0,1           ;"GOTO" OR "GOSUB" ?
    33660   2303          JMP     @.EDEX        ;   NO
    33661   2401          JMP     @.+1          ;   YES
    33662  34576          EDEXG


    33663  33273 .EDTL:   EDITL
    33664  33400 .EDTS:   EDITS
    33665  33416 .EDTO:   EDITO
    33666    352 .STYP:   STYPE
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE *******
    33667      43          43
    33670   34120          IF.
    33671   20776 EDITI:   LDA     0,.-2       ;"IF" STATEMENT
    33672   40347          STA     0,NTES
    33673   30220          LDA     2,C114
    33674     753          JMP     NCOD1

    33675   20026 NCODD:   LDA     0,C6        ;"DATA" STATEMENT
    33676  126400          SUB     1,1
    33677    6120          DECIMAL             ;DECIMAL INPUT
    33700   20232          LDA     0,C245
    33701  142414          SEQ     2,0         ;"%" ?
    33702     413          JMP     NCOD2       ;  NO
    33703   24164          LDA     1,DAC       ;  YES
    33704   34172          LDA     3,DBC
    33705  175015          SNZ     3,3
    33706  125234          MOVZR#  1,1,SZR
    33707   20160          LDA     0,DA
    33710   24051          LDA     1,C100
    33711  101304          MOVS    0,0,SZR
    33712  106433          SLE     0,1
    33713  103120          ADDZL   0,0
    33714  122401          SUB     1,0,SKP
    33715   20051 NCOD2:   LDA     0,C100
    33716   34350          LDA     3,.STMT
    33717  165400          INC     3,1
    33720   44333          STA     1,TSE
    33721   25400          LDA     1,0,3
    33722  107000          ADD     0,1
    33723   45400          STA     1,0,3
    33724  103120          ADDZL   0,0
    33725  101300          MOVS    0,0
    33726  101400          INC     0,0
    33727   40334          STA     0,TSE1
    33730   20232          LDA     0,C245
    33731  142414          SEQ     2,0
    33732     410          JMP     NCOD3+4
    33733    6125          INBYTE
    33734   20236          LDA     0,C254
    33735  142414          SEQ     2,0
    33736   20026 NCOD3:   LDA     0,C6
    33737  126400          SUB     1,1
    33740    6120          DECIMAL
    33741   50335          STA     2,TSE2
    33742  102400          SUB     0,0
    33743   24334          LDA     1,TSE1
    33744   30333          LDA     2,TSE
    33745  155000          MOV     2,3
    33746  137000          ADD     1,3
    33747   54333          STA     3,TSE
    33750    6120          DECIMAL
```

```
;                          << SI = R94PROTECTSC;  BD = 1/A.SPP4.9062!  >>
;  ******** THIS PAGE IS GARBAGE ******
    33751   12350          ISZ     @.STMT
    33752   30335          LDA     2,TSE2
    33753   20333          LDA     0,TSE
    33754   24210          LDA     1,.ESSE
    33755  106033          SLS     0,1
    33756     404          JMP     .+4
    33757   20236          LDA     0,C254
    33760  142415          SNE     2,0
    33761     755          JMP     NCOD3
    33762   20055          LDA     0,C215
    33763  142414          SEQ     2,0
    33764   20333          LDA     0,TSE
    33765  101120          MOVZL   0,0
    33766   40353          STA     0,EBP
    33767    2307          JMP     @.EDXE

    33770  102400 EDITC:   SUB     0,0          ;NOT A STATEMENT
    33771   24032          LDA     1,C12
    33772  152400          SUB     2,2
    33773    6325          JSR     @.SWTB       ;KEYBOARD COMMAND ?
    33774    6330          NCODERROR            ;   NO, UNRECOGNIZABLE WORD
    33775  104410          11*K!NOP
    33776    4413          JSR     NCOD4        ;   YES
    33777   34027          RUNC
    34000   37013          LIST
    34001   32650          SIZE
    34002   32517          NEW
    34003   35664          RNUM
    34004   36753          TAPE
    34005   32774          HELP
    34006   33470          LOAD
    34007   36756          DUMP
    34010   33425          DLTE
    34011  157000 NCOD4:   ADD     2,3          ;BRANCH ON KEYBOARD COMMAND
    34012   30355          LDA     2,.BUS
    34013   21024          LDA     0,FLAG.,2
    34014   24042          LDA     1,C20
    34015  107414          AND#    0,1,SZR      ;CHANNEL LOAD ?
    34016    6330          NCODERROR            ;   YES, CAN'T EXECUTE
    34017  137410          77*K!NOP
    34020   30005          LDA     2,RUP        ;   NO
    34021   21002          LDA     0,FBA.,2
    34022   41005          STA     0,OBP.,2
    34023    7400          JSR     @0,3         ;TABLE BRANCH
    34024    2306          JMP     @.EDUN
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
    34025  32204 BIPPY: BPS+4
;******* BELOW THIS LINE IS GARBAGE *******
    34026 100376        @376
    34027  20346 RUNC:  LDA     0,LNO      ;"RUN" COMMAND
    34030  24775         LDA     1,.-3      ;ASSUME SIMPLE RUN COMMAND
    34031 101015         SNZ     0,0        ;[LINE #] "RUN" ?
    34032    417         JMP     RUNC1      ;   NO
    34033   6277         JSR     @.CPYP     ;   YES
    34034  20346         LDA     0,LNO
    34035   6321         JSR     @.SFSN     ;FIND STARTING STATEMENT
    34036  34355         LDA     3,.BUS
    34037 172400         SUB     3,2        ;CONVERT TO DISPLACEMENT
    34040  51402         STA     2,PLC.,3
    34041   6101         CALL
    34042 100011         WONA
    34043   6133         OUTTEXT            ;CARRIAGE RETURN
    34044 106400         215*K
    34045   6141         STOUTPUT
    34046   6101         CALL
    34047 100011         WONA
    34050  24756         LDA     1,RUNC-1   ;ENTRY TO "RUN" AT SPECIFIED LINE



;****** ABOVE THIS LINE IS GARBAGE *******
    34051  34355 RUNC1: LDA     3,.BUS
    34052  21424         LDA     0,FLAG.,3
    34053  30273         LDA     2,MFLAG
    34054 143400         AND     2,0        ;CLEAR "EXECUTE MODE" FLAG
    34055  41424         STA     0,FLAG.,3
    34056  34005         LDA     3,RUP
    34057  20200         LDA     0,D.SAVE   ;CHANGE PROCESSOR TO "SAVE"
    34060  35425         LDA     3,DFT.,3
    34061  41771         STA     0,FDA+CHM1,3
    34062  34743         LDA     3,BIPPY
    34063   2117         JMP     @BUMPU&377;BUMP TO "SAVE"
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;********* THIS PAGE IS GARBAGE ******
          34064 WTBL=.  ;WORD TABLE FOR ENCODE AND DECODE

     34064 145256 145256 ;RUN              KEYBOARD COMMANDS
     34065  30463 030463 ;LIST
     34066  46472 046472 ;SIZE
     34067 134267 134267 ;NEW
     34070  44256 044256 ;RENUMBER
     34071  50060 050060 ;TAPE
     34072  20254 020254 ;HELP
     34073  30741 030741 ;LOAD
     34074  11255 011255 ;DUMP
     34075  10254 010254 ;DELETE

     34076 114700 114700 ;FN      100     USER-DEFINED FUNCTIONS (101-132)

     34077 150740 150740 ;TO      344     "OPERATOR" WORDS
     34100 150042 150042 ;TAB     345
     34101  47205 047205 ;STEP    346
     34102  53151 053151 ;USING   347

     34103  50405 050405 ;THEN    100     STATEMENT DIRECTIVE WORDS
     34104  16764 016764 ;GOTO    100
     34105  16763 016763 ;GOSUB   101
     34106  46447 046447 ;SIGNAL  102
     34107  26454 026454 ;KILL    103
     34110   5251 005251 ;BUILD   104
     34111  37005 037005 ;OPEN    105
     34112   6617 006617 ;CLOSE   106
     34113  46241 046241 ;SEARCH  107
     34114  44056 044056 ;RANDOM  110
     34115   6401 006401 ;CHAIN   111
     34116   6054 006054 ;CALL    112
     34117      0 000000 ;SPARE   113
     34120 122300 IF.:122300 ;IF 114
     34121 136700 136700 ;ON      115
     34122  14456 014456 ;FIND    116
     34123  47217 047217 ;STOP    117
     34124 110246 110246 ;DEF     120
     34125 112704 112704 ;END     121
     34126  44264 044264 ;RETURN  122
     34127 114762 114762 ;FOR     123
     34130  34270 034270 ;NEXT    124
     34131  10064 010064 ;DATA    125
     34132 144255 144255 ;REM     126
     34133 130264 130264 ;LET     127
     34134 132064 132064 ;MAT     130
     34135 110455 110455 ;DIM     131
     34136  44263 044263 ;RESTOR  132
     34137  22720 022720 ;INPUT   133     \
     34140  41111 041111 ;PRINT   134      ) ALSO USED AS
     34141  44241 044241 ;READ    135      ) MAT FUNCTIONS
     34142  57111 057111 ;WRITE   136     /
```

```
;                          << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE *******
    34143 113122 113122 ;ERR     137     NUMERIC FUNCTIONS
    34144 122724 122724 ;INT     140
    34145 102123 102123 ;ABS     141
    34146 146356 146356 ;SGN     142
    34147 144704 144704 ;RND     143
    34150 147062 147062 ;SQR     144
    34151 130747 130747 ;LOG     145
    34152 113420 113420 ;EXP     146
    34153 146456 146456 ;SIN     147
    34154 106763 106763 ;COS     150
    34155 103216 103216 ;ATN     151
    34156 150056 150056 ;TAN     152
    34157 110264 110264 ;DET     153
    34160 110326 110326 ;DFV     154     DUMMY FUNCTIONS
    34161 110301 110301 ;DFA     155
    34162 110320 110320 ;DFP     156
    34163 115101 115101 ;FRA     157     SPECIAL FUNCTIONS
    34164 132056 132056 ;MAN     160
    34165 106422 106422 ;CHR     161
    34166 123422 123422 ;IXR     162
    34167 147003 147003 ;SPC     163
    34170 134764 134764 ;NOT     164
    34171 106406 106406 ;CHF     165
    34172 130256 130256 ;LEN     166
    34173 151116 151116 ;TRN     167     MAT FUNCTIONS
    34174 164262 164262 ;ZER     170
    34175 106756 106756 ;CON     171
    34176 122216 122216 ;IDN     172
    34177 122726 122726 ;INV     173

              114 WTL =.-WTBL   ; WORD TABLE LENGTH
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE *******
    34200       0 000000 ;RUN                SECTION 2 OF WTBL
    34201      24 000024 ;LIST
    34202       5 000005 ;SIZE
    34203       0 0
    34204   52642 052642 ;RENUMBER
    34205       5 000005 ;TAPE
    34206      20 000020 ;HELP
    34207       4 000004 ;LOAD
    34210      20 000020 ;DUMP
    34211   13205 013205 ;DELETE
    34212       0 0
    34213       0 0
    34214       0 0
    34215      20 000020 ;STEP
    34216   34340 034340 ;USING
    34217      16 000016 ;THEN
    34220      17 000017 ;GOTO
    34221   52100 052100 ;GOSUB
    34222   34054 034054 ;SIGNAL
    34223      14 000014 ;KILL
    34224   30200 030200 ;BUILD
    34225      16 000016 ;OPEN
    34226   46240 046240 ;CLOSE
    34227   44150 044150 ;SEARCH
    34230   10755 010755 ;RANDOM
    34231   22700 022700 ;CHAIN
    34232      14 000014 ;CALL
    34233       0 0
    34234       0 0
    34235       0 0
    34236       4 000004 ;FIND
    34237  100020 100020 ;STOP
    34240       0 0
    34241   34717 .EDXR:  EDEXR
    34242  153116 153116 ;RETURN
    34243       0 0
    34244  100024 100024 ;NEXT
    34245       1 000001 ;DATA
    34246       0 0
    34247       0 0
    34250   34701 .EDXL:  EDEXL
    34251       0 0
    34252  150762 150762 ;RESTOR
    34253   53200 053200 ;INPUT
    34254   35200 035200 ;PRINT
    34255       4 000004 ;READ
    34256   50240 050240 ;WRITE

                ; END OF WORD TABLE

                .EOT  ; "PROTECT" R8.0 SOURCE #2
```

```
;                    << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;  "BUSINESS PROTECT" SOURCE #3 OF 6 TAPES FOR "IRIS" R8.0
;********* THIS PAGE IS GARBAGE *******

 34257  20346 EDEXK: LDA     0,LNO      ;CONSTANT > 40 OR NON-INTEGER
 34260 101015         SNZ     0,0        ;EXECUTE MODE ?
 34261    414         JMP     EDXK1      ;   YES, USE FLOATING POINT
 34262  20024         LDA     0,C4       ;   NO
 34263 122432         SGR     1,0        ;MORE THAN 4 DIGITS . . .
 34264 175014         SKZ     3,3        ;OR NON-INTEGER ?
 34265    410         JMP     EDXK1      ;   YES
 34266 122414         SEQ     1,0        ;EXACTLY 4 DIGITS ?
 34267    404         JMP     .+4        ;   NO
 34270  20160         LDA     0,DA       ;   YES
 34271 101112         SSP     0,0        ;FIRST DIGIT > 7 ?
 34272    403         JMP     EDXK1      ;   YES
 34273 126400         SUB     1,1        ;   NO
 34274    413         JMP     EDXK2      ;STORE AS ONE-WORD INTEGER

 34275 167000 EDXK1:  ADD     3,1        ;MUST STORE IN FLOATING FORM
 34276  20026         LDA     0,C6
 34277 122433         SLE     1,0        ;MORE THAN 6 DIGITS ?
 34300    403         JMP     .+3
 34301 126520         SUBZL   1,1        ;   NO, 2-WORD FLOATING
 34302    405         JMP     EDXK2
 34303  34032         LDA     3,C12      ;   YES
 34304 136433         SLE     1,3        ;MORE THAN 10 DIGITS ?
 34305 105221         MOVZR   0,1,SKP    ;   YES, 4-WORD FLOATING
 34306  24002         LDA     1,C2       ;   NO, 3-WORD FLOATING
 34307  44333 EDXK2: STA     1,TSE
 34310  30226         LDA     2,C174     ;"CONSTANT FOLLOWS" CODE
 34311 133000         ADD     1,2        ;ADD NUMBER TYPE
 34312   6332         SAVEBYTE
 34313  30353         LDA     2,EBP      ;NEXT BYTE ADDRESS
 34314 151620         INCZR   2,2        ;NEXT WORD ADDRESS
 34315  20333         LDA     0,TSE
 34316 105004         MOV     0,1,SZR    ;ONE-WORD INTEGER ?
 34317 125400         INC     1,1        ;   NO
 34320 143000         ADD     2,0        ;NEXT WORD ADDRESS
 34321 101520         INCZL   0,0        ;NEXT BYTE ADDRESS
 34322  40353         STA     0,EBP
 34323 102400         SUB     0,0        ;STORE THE NUMBER
 34324   6120         DECIMAL
 34325    461         JMP     EDEX1


 34326  20052 EDEX2: LDA     0,C177     ;NOT A LETTER OR DIGIT
 34327 142433         SLE     2,0        ;FROM CARD READER ?
 34330    462         JMP     EDEX3      ;   NO
 34331 133000         ADD     1,2        ;   NO, ADD 301 AND STORE
 34332    563         JMP     EDEXS
 34333  30042         LDA     2,C20      ;   YES
 34334   2401         JMP     @.+1
 34335  34546         EDXW1
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;********* THIS PAGE IS GARBAGE ******
    34336    6125 EDEX:  INBYTE              ;ENCODE EXPRESSION
    34337   20237        LDA     0,C256
    34340  142414        SEQ     2,0         ;"." ?
    34341     410        JMP     .+10        ;   NO
    34342   34005        LDA     3,RUP       ;   YES
    34343   25404        LDA     1,IBP.,3    ;SCAN PAST SPACES
    34344  125400        INC     1,1
    34345    6124        GETBYTE
    34346   20056        LDA     0,C240
    34347  142415        SNE     2,0
    34350     774        JMP     .-4
    34351    6127        ISA2DIGIT           ;DIGIT ?
    34352     435        JMP     EDEX1+1     ;   NO
    34353   34005        LDA     3,RUP       ;   YES, LITERAL CONSTANT
    34354   15404        DSZ     IBP.,3
    34355   20026        LDA     0,C6        ;CONVERT NUMBER INPUT
    34356  126400        SUB     1,1
    34357    6120        DECIMAL
    34360    6142        TRAPFAULT           ;IT WAS TOO A NUMBER !?
    34361   50334        STA     2,TSE1      ;TERMINATOR CODE
    34362  105000        MOV     0,1         ;DIGIT COUNT
    34363   20164        LDA     0,DAC
    34364   34172        LDA     3,DBC       ;FRACTIONAL DIGIT COUNT
    34365  106414        SEQ     0,1         ;INTEGER ?
    34366     707        JMP     EDXK1       ;   NO
    34367   30002        LDA     2,C2        ;   MAYBE
    34370  132432        SGR     1,2         ;MORE THAN 2 DIGITS . . .
    34371  175014        SKZ     3,3         ;OR NON-INTEGER ?
    34372     665        JMP     EDEXK       ;   YES
    34373   30160        LDA     2,DA        ;   NO, INTEGER < 100
    34374  151300        MOVS    2,2
    34375  125234        MOVZR#  1,1,SZR
    34376     404        JMP     .+4
    34377  153120        ADDZL   2,2
    34400  153120        ADDZL   2,2
    34401  151300        MOVS    2,2
    34402   20051        LDA     0,C100
    34403  142433        SLE     2,0         ;VALUE > 40 ?
    34404     653        JMP     EDEXK       ;   YES
    34405    6332        SAVEBYTE            ;   NO, STORE ONE BYTE CONSTANT
    34406   30334 EDEX1: LDA     2,TSE1      ;CONTINUE SCAN
    34407    6130        ISA2LETTER          ;LETTER ?
    34410     716        JMP     EDEX2       ;   NO
    34411     447        JMP     EDEXV       ;   YES
```

```
;                          << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE ******
    34412   24234 EDEX3: LDA     1,C250    ;NOT A LETTER OR DIGIT
    34413  146415         SNE     2,1       ;"(" ?
    34414    2634         JMP     @.EDXL    ;   YES
    34415   24245         LDA     1,C333    ;   NO
    34416  146415         SNE     2,1       ;"[" ?
    34417    2631         JMP     @.EDXL    ;   YES
    34420   24246         LDA     1,C335    ;   NO
    34421  146415         SNE     2,1       ;"]" ?
    34422    2617         JMP     @.EDXR    ;   YES
    34423   24235         LDA     1,C251    ;   NO
    34424  146415         SNE     2,1       ;")" ?
    34425    2614         JMP     @.EDXR    ;   YES
    34426  146432         SGR     2,1       ;POSSIBLE RELATIONAL OR "^" ?
    34427     407         JMP     EDX3A     ;   NO
    34430   24243         LDA     1,C276    ;   MAYBE
    34431  146432         SGR     2,1
    34432    2465         JMP     @.EDXO    ;   YES
    34433   24247         LDA     1,C336    ;   NO
    34434  146415         SNE     2,1       ;"^" ?
    34435    2312         JMP     @.EDXX    ;   YES
    34436   24233 EDX3A: LDA     1,C247
    34437  146415         SNE     2,1       ;SINGLE QUOTE ?
    34440    2313         JMP     @.ESQ     ;   YES
    34441   24062         LDA     1,C300
    34442  146414         SEQ     2,1       ;"@" ?
    34443     403         JMP     EDX3B     ;   NO
    34444   30264         LDA     2,C365    ;   YES
    34445     450         JMP     EDEXS
    34446   24231 EDX3B: LDA     1,C242    ;   NO
    34447  146415         SNE     2,1       ;QUOTE ?
    34450    2310         JMP     @.EDXQ    ;   YES
    34451   34055         LDA     3,C215    ;   NO
    34452  156414         SEQ     2,3       ;RETURN ?
    34453    2445         JMP     @.EDXC    ;   NO
    34454   30251         LDA     2,C342    ;   YES
    34455    6332         SAVEBYTE          ;STORE END OF STATEMENT CODE
    34456     552         JMP     EDEXD
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062!  >>
;******* THIS PAGE IS GARBAGE *******
    34457   40000                40000
    34460   50334 EDEXV:STA      2,TSE1      ;VARIABLE OR FUNCTION WORD
    34461    6125                INBYTE
    34462    6130                ISA2LETTER  ;ANOTHER LETTER ?
    34463     402                SKIP
    34464     435                JMP     EDEXW       ;   YES, MAY BE A WORD
    34465   20334                LDA     0,TSE1      ;   NO, MUST BE A VARIABLE
    34466  101300                MOVS    0,0
    34467   40334                STA     0,TSE1      ;FIRST LETTER IN TOP BYTE
    34470   24057                LDA     1,C244
    34471  146415                SNE     2,1         ;"$" ?
    34472     417                JMP     EDEXS-4     ;   YES
    34473    6127                ISA2DIGIT           ;DIGIT ?
    34474     411                JMP     .+11        ;   NO, SINGLE LETTER NAME
    34475   24053                LDA     1,C200      ;   YES
    34476  132400                SUB     1,2
    34477  143000                ADD     2,0
    34500   40334                STA     0,TSE1
    34501    6125                INBYTE              ;CHECK NEXT BYTE
    34502   20057                LDA     0,C244
    34503  142415                SNE     2,0         ;"$" ?
    34504     405                JMP     .+5         ;   YES
    34505   34005                LDA     3,RUP       ;   NO, BACK UP
    34506   15404                DSZ     IBP.,3
    34507   24334                LDA     1,TSE1
    34510     404                JMP     .+4

    34511   24334                LDA     1,TSE1      ;STRING VARIABLE
    34512   20745                LDA     0,EDEXV-1;SET FLAG IN NAME WORD
    34513  107000                ADD     0,1
    34514    6322                JSR     @.SFVN      ;CONVERT TO VARIABLE NUMBER

    34515    6332 EDEXS:SAVEBYTE             ;STORE EDITED BYTE
    34516     620                JMP     EDEX
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;********** THIS PAGE IS GARBAGE *******
    34517  34777 .EDXO:EDEXO
    34520  35156 .EDXC:EDEXC


    34521   6276 EDEXW: JSR     @.COMB    ;WORD
    34522  34327        LDA     3,.WTBL
    34523   2452        JMP     @.EDXT    ;   YES
    34524  25434        LDA     1,IF.-WTBL,3
    34525  21435        LDA     0,IF.-WTBL+1,3
    34526 132535        SUBZL#  1,2,SNR   ;"IF" ?
    34527    415        JMP     EDXW1-2   ;   YES
    34530 112535        SUBZL#  0,2,SNR   ;"ON" ?
    34531    414        JMP     EDXW1-1   ;   YES
    34532  20034        LDA     0,C14     ;   NO
    34533  24347        LDA     1,NTES
    34534   6325        JSR     @.SWTB    ;SEARCH WORD TABLE
    34535   2437        JMP     @.EDXN    ;   NOT FOUND
    34536  24037        LDA     1,C17
    34537 146415        SNE     2,1       ;"THEN" ?
    34540 151400        INC     2,2       ;   YES, CHANGE TO "GOTO"
    34541 146432        SGR     2,1       ;KEYBOARD COMMAND ?
    34542    426        JMP     EDXW2     ;   YES
    34543    403        JMP     EDXW1     ;   NO, STATEMENT DIRECTIVE

    34544 152001        ADC     2,2,SKP
    34545 152400        SUB     2,2
    34546  50333 EDXW1: STA     2,TSE     ;STATEMENT FOLLOWS
    34547  30252        LDA     2,C343
    34550   6332        SAVEBYTE
    34551 151400        INC     2,2
    34552  50350        STA     2,.STMT
    34553 102400        SUB     0,0
    34554  42350        STA     0,@.STMT
    34555 151120        MOVZL   2,2
    34556  50353        STA     2,EBP
    34557  30333        LDA     2,TSE
    34560 151415        INC#    2,2,SNR
    34561   2404        JMP     @.+4
    34562 151015        SNZ     2,2
    34563   2403        JMP     @.+3
    34564   2403        JMP     @.+3
    34565  33671        EDITI
    34566  33416        EDITO
    34567  33645        EDITK

    34570  24244 EDXW2: LDA     1,C331
    34571 133000        ADD     1,2
    34572    723        JMP     EDEXS


    34573  34725 .EDXF:EDEXF
    34574  34767 .EDXN:EDEXN
    34575  34763 .EDXT:EDEXT
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE ******
    34576   24353 EDEXG: LDA     1,EBP      ;GOTO OR GOSUB
    34577  125620        INCZR   1,1
    34600   44334        STA     1,TSE1
    34601    6323 EDEX5: JSR     @.SNI      ;GET LINE NUMBER
    34602  101015        SNZ     0,0        ;ZERO ?
    34603   42334        STA     0,@TSE1    ;   NO, STORE IN EBS
    34604   10334        ISZ     TSE1
    34605   12350        ISZ     @.STMT     ;INCREMENT LNO COUNT
    34606   20344        LDA     0,ONF
    34607  101015        SNZ     0,0        ;"ON" STATEMENT ?
    34610     404        JMP     .+4        ;   NO
    34611   20236        LDA     0,C254     ;   YES
    34612  142415        SNE     2,0        ;COMMA ?
    34613     766        JMP     EDEX5      ;   YES, GET ANOTHER LINE NUMBER
    34614   24334        LDA     1,TSE1     ;   NO
    34615  125120        MOVZL   1,1
    34616   44353        STA     1,EBP
    34617   24055        LDA     1,C215
    34620  146415        SNE     2,1        ;END OF LINE ?
    34621     407        JMP     EDEXD      ;   YES
    34622   24056        LDA     1,C240     ;   NO
    34623  132014        ADC#    1,2,SZR    ;"!" ?
    34624    2401        JMP     @.+1       ;   YES
    34625   33257        EDITR-2


    34626     102        102
    34627     123        123
    34630   30345 EDEXD: LDA     2,SSC
    34631   20207        LDA     0,.ESS
    34632  142404        SUB     2,0,SZR
    34633     403        JMP     .+3
    34634   25000        LDA     1,0,2
    34635  106014        ADC#    0,1,SZR
    34636    6330        NCODERROR          ;INCORRECT () CLOSURE
    34637  105410        13*K!NOP
    34640   30076 EDEXE: LDA     2,ERRF     ;END OF EXPRESSION
    34641  151014        SKZ     2,2
    34642    6330        NCODERROR
    34643  107410        17*K!NOP
    34644   24353        LDA     1,EBP
    34645  125212        MOVR#   1,1,SZC    ;ON ODD BYTE BOUNDARY ?
    34646    6134        PUTBYTE            ;   YES, FILL WITH ZERO
    34647   20346        LDA     0,LNO
    34650  101014        SKZ     0,0        ;LINE NUMBER GIVEN ?
    34651    6304        JSR     @.EDML     ;   YES
    34652   20351        LDA     0,NEXF     ;   NO, SET UP TO EXECUTE
    34653   24352        LDA     1,STYPE
    34654   30752        LDA     2,EDEXD-2
    34655  132433        SLE     1,2        ;"GOTO", "GOSUB", OR "SIGNAL" . . .
    34656  101014        SKZ     0,0        ;OR "NO EXECUTE" FLAG SET ?
    34657     407        JMP     EDXEE      ;   YES
    34660   30747        LDA     2,EDEXD-1  ;   NO
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;********** THIS PAGE IS GARBAGE ******
    34661 132014          ADC#    1,2,SZR   ;"RETURN" . . .
    34662 146015          ADC#    2,1,SNR   ;OR "NEXT" ?
    34663    403          JMP     EDXEE     ;   YES
    34664  30217          LDA     2,C112
    34665 132015          ADC#    1,2,SNR   ;CHAIN ?
    34666   6330 EDXEE:   NCODERROR           ;   YES, ILLEGAL IN EXECUTE MODE
    34667 133410          67*K!NOP
    34670  34355          LDA     3,.BUS    ;   NO
    34671  25424          LDA     1,FLAG.,3
    34672  20042          LDA     0,C20
    34673 123414          AND#    1,0,SZR   ;LOAD MODE ?
    34674   6330          NCODERROR           ;   YES, EXECUTE IS ILLEGAL
    34675 140010          100*K!NOP
    34676 126400          SUB     1,1       ;   NO, ENCODE AS LINE # 0
    34677  44346          STA     1,LNO
    34700   6304          JSR @.EDML

    34701  30262 EDEXL:   LDA     2,C362    ; "(" OR "["
    34702  24343          LDA     1,LBS
    34703  20053          LDA     0,C200
    34704 122432          SGR     1,0
    34705    404          JMP     .+4
    34706  20250          LDA     0,C340
    34707 122032          SGE     1,0
    34710    404          JMP     .+4
    34711  12345          ISZ     @SSC
    34712   2311          JMP     @.EDXS
    34713    723          JMP     EDEXE-2

    34714  10345          ISZ     SSC
    34715  20345          LDA     0,SSC
    34716  24210          LDA     1,.ESSE
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE ******

34717  30265 EDEXR: LDA     2,C366    ;  ")" OR "]"
34720  16345        DSZ     @SSC
34721   2311        JMP     @.EDXS
34722  14345        DSZ     SSC
34723 151400 EDXRX: INC     2,2
34724   2311        JMP     @.EDXS

34725   6274 EDEXF: JSR     @.ACNL    ;USER FUNCTION
34726  24051        LDA     1,C100
34727 133000        ADD     1,2
34730  24343        LDA     1,LBS
34731  34222        LDA     3,C121
34732 136014        ADC#    1,3,SZR
34733   2311        JMP     @.EDXS
34734  24353        LDA     1,EBP
34735  34350        LDA     3,.STMT
34736 175140        MOVOL   3,3
34737 136414        SEQ     1,3
34740   2311        JMP     @.EDXS
34741   6332        SAVEBYTE
34742   6125        INBYTE
34743  24234        LDA     1,C250
34744 146414        SEQ     2,1
34745   6274        JSR     @.ACNL
34746 141300        MOVS    2,0
34747  34355        LDA     3,.BUS
34750  25401        LDA     1,VDT.,3
34751 137000        ADD     1,3
34752  41400        STA     0,0,3
34753  24062        LDA     1,C300
34754 133000        ADD     1,2
34755   6332        SAVEBYTE
34756   6125        INBYTE
34757  24235        LDA     1,C251
34760 146414        SEQ     2,1
34761   2303        JMP     @.EDEX

34762 133001        ADD     1,2,SKP
34763  30253 EDEXT: LDA     2,C344    ;"TO"
34764   2311        JMP     @.EDXS

34765     53        53
34766     42        42
34767  20776 EDEXN: LDA     0,.-2     ;WORD NOT AN OPERATOR
34770  24776        LDA     1,.-2
34771 152400        SUB     2,2
34772   6325        JSR     @.SWTB
34773    545        JMP     EDEXB
34774  24213        LDA     1,C60
34775    765        JMP     EDEXT-1
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;********** THIS PAGE IS GARBAGE *******
    34776   35643          EDTBL-252
    34777   20240 EDEXO:LDA     0,C272     ;OPERATOR SYMBOL
    35000  142433          SLE     2,0        ;ARITHMETIC OPERATOR ?
    35001     420          JMP     EDEX7      ;   NO
    35002  142414          SEQ     2,0        ;   YES, ":" ?
    35003     403          JMP     EDXO1      ;     NO
    35004   30271          LDA     2,C376     ;     YES, SAVE CODE FOR IT
    35005    2311          JMP     @.EDXS
    35006   20770 EDXO1:LDA     0,EDEXO-1
    35007  113000          ADD     0,2
    35010   31000          LDA     2,0,2      ;GET INTERNAL CODE
    35011   20261          LDA     0,C361
    35012  142414          SEQ     2,0        ;"," ?
    35013    2311          JMP     @.EDXS     ;   NO
    35014   20345          LDA     0,SSC      ;   YES
    35015   24207          LDA     1,.ESS
    35016  106414          SEQ     0,1        ;WITHIN SUBSCRIPTS ?
    35017   30263          LDA     2,C364     ;   YES, SUBSTITUTE OTHER COMMA
    35020    2311          JMP     @.EDXS

    35021  145000 EDEX7:MOV     2,1        ;RELATIONAL OPERATOR OR ";"
    35022   30256          LDA     2,C352
    35023   20242          LDA     0,C275
    35024   34343          LDA     3,LBS
    35025  106014          ADC#    0,1,SZR    ;THIS BYTE ">" ?
    35026     406          JMP     .+6        ;   NO
    35027   20260          LDA     0,C357     ;   YES
    35030  162415          SNE     3,0        ;PREVIOUS BYTE "<" ?
    35031     417          JMP     EDEX8      ;   YES, "<>"
    35032  151400          INC     2,2        ;   NO, ">"
    35033    2311          JMP     @.EDXS

    35034  151400          INC     2,2
    35035  151400          INC     2,2
    35036  122414          SEQ     1,0        ;THIS BYTE "=" ?
    35037     413          JMP     EDEX8+2    ;   NO
    35040   20257          LDA     0,C353     ;   YES
    35041  162415          SNE     3,0        ;PREVIOUS BYTE ">" ?
    35042     406          JMP     EDEX8      ;   YES, ">="
    35043  151400          INC     2,2        ;   NO
    35044   20260          LDA     0,C357
    35045  162414          SEQ     3,0        ;PREVIOUS BYTE "<" ?
    35046    2311          JMP     @.EDXS     ;   NO, "="
    35047  151400          INC     2,2        ;   YES, "<="
    35050   14353 EDEX8:DSZ     EBP        ;BACK UP BYTE POINTER
    35051    2311          JMP     @.EDXS     ;STORE DOUBLE SYMBOL CODE

    35052   30260          LDA     2,C357     ;ASSUME "<"
    35053  122014          ADC#    1,0,SZR    ;IS IT ?
    35054  151400          INC     2,2        ;   NO, MUST BE ";"
    35055    2311          JMP     @.EDXS     ;STORE SINGLE SYMBOL CODE

    35056  152401 EDEXZ:SUB     2,2,SKP    ;STORE END OF STRING
    35057   30267 EDEXX:LDA     2,C374     ;STORE "^"
    35060    2311          JMP     @.EDXS
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE *******
    35061       2           .BLK 2
    35063   30255 EDEXQ:LDA   2,C351    ;BEGIN LITERAL STRING
    35064    6332           SAVEBYTE
    35065    6126           INSTBYTE
    35066   24231           LDA     1,C242    ;   NO
    35067  146415           SNE     2,1       ;END OF STRING ?
    35070     766           JMP     EDEXZ     ;   YES
    35071   24234           LDA     1,C250    ;   NO
    35072  146014           ADC#    2,1,SZR   ;APOSTROPHE ?
    35073     407           JMP     .+7       ;   NO
    35074   20343           LDA     0,LBS     ;   YES
    35075  106014           ADC#    0,1,SZR   ;LAST BYTE ALSO AN APOSTROPHE ?
    35076     766           JMP     EDEXQ+1   ;   NO
    35077   14353           DSZ     EBP       ;   YES, REPLACE WITH A QUOTE
    35100   30231           LDA     2,C242
    35101     763           JMP     EDEXQ+1

    35102   24230           LDA     1,C232
    35103  146414           SEQ     2,1       ;CONTROL Z ?
    35104     403           JMP     .+3
    35105   30055           LDA     2,C215    ;   YES, STORE A RETURN
    35106     756           JMP     EDEXQ+1
    35107   20063           LDA     0,C334    ;   NO
    35110  142404           SUB     2,0,SZR   ;BACKSLASH ?
    35111     753           JMP     EDEXQ+1   ;   NO
    35112   24024           LDA     1,C4      ;   YES
    35113   44747           STA     1,EDEXQ-1
    35114   40745 EDXQ1:STA 0,EDEXQ-2
    35115    6125           INBYTE            ;GET A CONTROL CHARACTER BYTE
    35116    6127           ISA2DIGIT         ;DIGIT ?
    35117     412           JMP     EDXQ2     ;   NO
    35120   14742           DSZ     EDEXQ-1   ;   YES
    35121  132401           SUB     1,2,SKP   ;CONVERT TO BINARY
    35122   34027           LDA     3,C7
    35123  156433           SLE     2,3       ;LEGAL DIGIT ?
    35124   20735           LDA     0,EDEXQ-2;   YES
    35125  103120           ADDZL   0,0       ;SHIFT LEFT
    35126  101120           MOVZL   0,0
    35127  143000           ADD     2,0
    35130     764           JMP     EDXQ1

    35131   24063 EDXQ2:LDA 1,C334
    35132  146414           SEQ     2,1       ;TERMINATING BACKSLASH ?
    35133   30726           LDA     2,EDEXQ-2;   YES
    35134   24053           LDA     1,C200
    35135  151015           SNZ     2,2       ;ZERO ENTERED ?
    35136  133000           ADD     1,2       ;   YES, SET MSB
    35137     725           JMP     EDEXQ+1
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE *******
    35140 102120 EDEXB: ADCZL   0,0         ;BACK UP 3 NON-SPACES
    35141  40335        STA     0,TSE2
    35142  34005        LDA     3,RUP
    35143  15404        DSZ     IBP.,3
    35144  15404        DSZ     IBP.,3
    35145   6126        INSTBYTE
    35146  20056        LDA     0,C240
    35147 142414        SEQ     2,0
    35150  10335        ISZ     TSE2
    35151    771        JMP     EDEXB+2
    35152  24043        LDA     1,C37
    35153 147700        ANDS    2,1
    35154   2401        JMP     @.+1
    35155  34514        EDEXS-1

    35156 132014 EDEXC: ADC#    1,2,SZR     ;CHECK FOR "#", DUMMY VARIABLE, OR "!"
    35157    404        JMP     .+4
    35160  30064        LDA     2,C377      ;"#", DISALLOW EXECUTING
    35161  50351        STA     2,NEXF
    35162   2311        JMP     @.EDXS

    35163  24214        LDA     1,C71       ;NOT "#"
    35164 133000        ADD     1,2
    35165  24247        LDA     1,C336
    35166 146414        SEQ     2,1         ;"%"
    35167 132015        ADC#    1,2,SNR     ;OR "&" ?
    35170   2311        JMP     @.EDXS      ;   YES, DUMMY VARIABLE
    35171  24245        LDA     1,C333      ;   NO
    35172 146014        ADC#    2,1,SZR     ;IS IT "!" ?
    35173   2401        JMP     @.+1        ;   YES, COPY THE COMMENT
    35174  33257        EDITR-2

                .EOT    ; "PROTECT" R9.0 SOURCE #3
```

```
;                    << SI = R94PROTECTSC;  BD = 1/A.SPP4.9062! >>
;  "BUSINESS PROTECT" SOURCE #4 OF 6 TAPES FOR "IRIS" R8.0

 35175  41177          EBS-1
 35176  54334 EDML:    STA     3,TSE1      ;ENCODE DONE, MOVE LINE
 35177  34355          LDA     3,.BUS
 35200 101015          SNZ     0,0         ;EMPTY LINE ?
 35201   2306          JMP     @.EDUN      ;   YES
 35202  21424          LDA     0,FLAG.,3   ;   NO
 35203  24042          LDA     1,C20
 35204 107414          AND#    0,1,SZR     ;IN LOAD MODE ?
 35205   6277          JSR     @.CPYP      ;   YES, CHECK PROTECTION
 35206  34355          LDA     3,.BUS
 35207 102400          SUB     0,0         ;CLEAR ERROR BRANCH
 35210  41452          STA     0,ERBP.,3
 35211  20353          LDA     0,EBP       ;B(LAST BYTE IN EDIT BUFFER) + 1
 35212 100400          NEG     0,0
 35213 100220          COMZR   0,0
 35214  40335          STA     0,TSE2      ;A(LAST WORD IN EDIT BUFFER)
 35215  31405          LDA     2,SLT.,3
 35216  25406          LDA     1,UVS.,3
 35217  21401          LDA     0,VDT.,3
 35220 173000          ADD     3,2         ;.SLT
 35221 117000          ADD     0,3         ;.VDT
 35222  45401          STA     1,1,3       ;DUMMY LAST STATEMENT "POINTER"
 35223  24346          LDA     1,LNO       ;LINE NUMBER ENTERED
 35224    403          JMP     .+3

 35225 151400          INC     2,2         ;SEARCH SLT FOR LINE NUMBER
 35226 151400          INC     2,2
 35227  21000          LDA     0,0,2
 35230 156032          SGE     2,3         ;END OF SLT . . .
 35231 106433          SLE     0,1         ;OR PAST DESIRED POSITION ?
 35232    410          JMP     EDMLI       ;   YES
 35233 106414          SEQ     0,1         ;LINE NUMBERS MATCH ?
 35234    771          JMP     .-7         ;   NO
 35235  20334          LDA     0,TSE1      ;   YES
 35236 101015          SNZ     0,0         ;DELETING A LINE ?
 35237    420          JMP     EDMLD       ;   YES
 35240 126400          SUB     1,1         ;   NO, FLAG "REPLACE"
 35241    405          JMP     EDMLI+4

 35242  20334 EDMLI:   LDA     0,TSE1      ;LINE NUMBER NOT FOUND
 35243 101015          SNZ     0,0         ;DELETING A LINE ?
 35244   4204          JSR     ERRLN
 35245  24002          LDA     1,C2        ;FLAG "INSERT"
 35246  34335          LDA     3,TSE2      ;<< ENTRY FOR "REPLACE"
 35247  20726          LDA     0,EDML-1
 35250 116400          SUB     0,3         ;SIZE OF LINE IN EBS (# WORDS)
 35251  54336          STA     3,TSE3
 35252 125015          SNZ     1,1         ;REPLACING A LINE ?
 35253    406          JMP     EDMLD+2     ;   YES
 35254 102400          SUB     0,0         ;   NO, OLD LINE SIZE = 0
 35255  35001          LDA     3,1,2       ;A(LINE AFTER INSERTION)
 35256    406          JMP     EDMLD+5
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
    35257  40336 EDMLD: STA     0,TSE3     ;NEW LINE SIZE = 0
    35260 126120         ADCZL   1,1        ;FLAG "DELETE"
    35261  35001         LDA     3,1,2      ;<< ENTRY FOR "REPLACE"
    35262  21003         LDA     0,3,2
    35263 162400         SUB     3,0        ;SIZE OF OLD LINE (# WORDS)
    35264  44334         STA     1,TSE1     ;<< ENTRY FOR "INSERT"
    35265  40337         STA     0,TSE4
    35266  50340         STA     2,TSE5
    35267  30355         LDA     2,.BUS
    35270 157000         ADD     2,3
    35271  54341         STA     3,TSE6
    35272  30336         LDA     2,TSE3
    35273 112400         SUB     0,2        ;STATEMENT SIZE DIFFERENCE (# WORDS)
    35274 133000         ADD     1,2
    35275  50342         STA     2,TSE7     ;TOTAL SIZE DIFFERENCE (# WORDS)

;AT THIS POINT:
;   TSE1 = +2(INSERT), 0(REPLACE), OR -2(DELETE)
;   TSE2 = ADDRESS OF LAST WORD IN EBS
;   TSE3 = NUMBER OF WORDS IN NEW LINE IN EBS
;   TSE4 = NUMBER OF WORDS IN OLD LINE IN UPS
;   TSE5 = POINTER FOR OLD LINE MOVE SOURCE
;   TSE6 = POINTER TO NEW LINE MOVE DESTINATION
;   TSE7 = TOTAL SIZE DIFFERENCE (# WORDS)

    35276  34355         LDA     3,.BUS
    35277  21407         LDA     0,NVS.,3
    35300 163000         ADD     3,0        ;CURRENT .NVS
    35301 100400         NEG     0,0        ; MINUS
    35302 100000         COM     0,0        ;  ONE
    35303 143000         ADD     2,0        ;CALCULATE NEW NVS POINTER
    35304  34356         LDA     3,.EUS
    35305 116433         SLE     0,3        ;ENOUGH SPACE ?
    35306   4202         JSR     ERRSO      ;  NO
    35307  34355         LDA     3,.BUS     ;  YES
    35310  31405         LDA     2,SLT.,3
    35311 173000         ADD     3,2
    35312  34340         LDA     3,TSE5     ;START ADJUSTING SLT & VDT POINTERS
    35313 156033 EDMLA:  SLS     2,3        ;END OF A PASS ?
    35314    410         JMP     EDMLB      ;  YES
    35315  21001         LDA     0,1,2      ;  NO, GET "POINTER"
    35316 101014         SKZ     0,0        ;LOCATION ASSIGNED ?
    35317 123000         ADD     1,0        ;  YES, ADJUST
    35320  41001         STA     0,1,2
    35321 151400         INC     2,2        ;<< ENTRY FROM EDMLB
    35322 151400         INC     2,2
    35323    770         JMP     EDMLA
```

```
;                        << SI = R94PROTECTSC;  BD = 1/A.SPP4.9062!  >>
    35324   34355 EDMLB: LDA     3,.BUS     ;START 2ND ADJUSTMENT PASS
    35325   21404        LDA     0,UPS.,3
    35326  117000        ADD     0,3        ;.UPS
    35327  156033        SLS     2,3        ;ALL POINTERS ADJUSTED ?
    35330     406        JMP     .+6        ;   YES
    35331  121000        MOV     1,0        ;   NO
    35332   24342        LDA     1,TSE7
    35333  101014        SKZ     0,0        ;REPLACING A LINE ?
    35334     757        JMP     EDMLA      ;   NO
    35335     764        JMP     EDMLA+6    ;   YES, BYPASS ONE POINTER

    35336   24334        LDA     1,TSE1
    35337  125113        SSN     1,1        ;DELETING A LINE ?
    35340     407        JMP     .+7        ;   NO
    35341   30340        LDA     2,TSE5     ;   YES, CLOSE UP SLT
    35342  141400        INC     2,0
    35343  101400        INC     0,0
    35344   24341        LDA     1,TSE6
    35345    6101        CALL
    35346  100015        MOVEWORDS
    35347   34355        LDA     3,.BUS     ;ADJUST SPACE IN UPS
    35350   25407        LDA     1,NVS.,3
    35351  167000        ADD     3,1
    35352   20341        LDA     0,TSE6
    35353  111000        MOV     0,2
    35354   34337        LDA     3,TSE4
    35355  163000        ADD     3,0
    35356   34336        LDA     3,TSE3
    35357  173000        ADD     3,2
    35360   34334        LDA     3,TSE1
    35361  173000        ADD     3,2
    35362  122432        SGR     1,0        ;ANYTHING TO BE MOVED ?
    35363     403        JMP     .+3        ;   NO
    35364    6101        CALL
    35365  100015        MOVEWORDS
    35366   24334        LDA     1,TSE1
    35367  124513        NEGL#   1,1,SNC    ;INSERTING A NEW LINE ?
    35370     420        JMP     EDMLN      ;   NO
    35371   20340        LDA     0,TSE5     ;   YES, EXPAND SLT
    35372   24341        LDA     1,TSE6
    35373  111400        INC     0,2
    35374  151400        INC     2,2
    35375    6101        CALL
    35376  100015        MOVEWORDS
    35377   30340        LDA     2,TSE5
    35400   20346        LDA     0,LNO      ;NEW LINE'S NUMBER
    35401   41000        STA     0,0,2
    35402   20341        LDA     0,TSE6     ;NEW LINE'S LOCATION
    35403   24334        LDA     1,TSE1
    35404   34355        LDA     3,.BUS
    35405  123000        ADD     1,0
    35406  162400        SUB     3,0
    35407   41001        STA     0,1,2      ;NEW LINE'S DISPLACEMENT
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
    35410 125112 EDMLN: SSP     1,1         ;INSERT OR REPLACE ?
    35411    407        JMP     .+7         ;   NO
    35412  30341        LDA     2,TSE6      ;   YES
    35413 133000        ADD     1,2
    35414  20206        LDA     0,.EBS      ;MOVE NEW LINE INTO UPS
    35415  24335        LDA     1,TSE2
    35416   6101        CALL
    35417 100015        MOVEWORDS
    35420  34355        LDA     3,.BUS      ;DO FINAL ADJUSTING
    35421  24334        LDA     1,TSE1
    35422  21401        LDA     0,VDT.,3    ;ADJUST .VDT
    35423 123000        ADD     1,0
    35424  41401        STA     0,VDT.,3
    35425  21404        LDA     0,UPS.,3    ;ADJUST .UPS
    35426 123000        ADD     1,0
    35427  41404        STA     0,UPS.,3
    35430  24342        LDA     1,TSE7
    35431  21406        LDA     0,UVS.,3    ;ADJUST .UVS
    35432 123000        ADD     1,0
    35433  41406        STA     0,UVS.,3
    35434  21407        LDA     0,NVS.,3    ;ADJUST .NVS
    35435 123000        ADD     1,0
    35436  41407        STA     0,NVS.,3
    35437  20341        LDA     0,TSE6      ;PREPARE TO ADJUST USER FUNC TABLE
    35440  30355        LDA     2,.BUS
    35441 142400        SUB     2,0         ;RELATIVE LOC OF NEW LINE
    35442  24342        LDA     1,TSE7      ;TOTAL DISPLACEMENT
    35443  34212        LDA     3,C32
    35444  54335        STA     3,TSE2      ;SIZE OF UFT
    35445  35011        LDA     3,UFT.,2
    35446 157000        ADD     2,3         ;.UFT
    35447  31400 EDML1: LDA     2,0,3       ;FUNCTION DEFINITION DISPLACEMENT
    35450 142032        SGE     2,0         ;IS IT AFFECTED ?
    35451    412        JMP     EDML2+2     ;   NO
    35452 142414        SEQ     2,0         ;WAS THIS DEFINITION CHANGED ?
    35453    405        JMP     EDML2-1     ;   NO
    35454  30334        LDA     2,TSE1      ;   YES
    35455 151112        SSP     2,2         ;WAS IT DELETED ?
    35456    403        JMP     EDML2       ;   YES, CLEAR POINTER
    35457  31400        LDA     2,0,3       ;   NO
    35460 133001        ADD     1,2,SKP     ;ADJUST POINTER
    35461 152400 EDML2: SUB     2,2
    35462  51400        STA     2,0,3
    35463 175400        INC     3,3         ;STEP UFT POINTER
    35464  14335        DSZ     TSE2        ;END OF UFT ?
    35465    762        JMP     EDML1       ;   NO
    35466  30355        LDA     2,.BUS      ;   YES
    35467  24340        LDA     1,TSE5
    35470 146400        SUB     2,1
    35471  44340        STA     1,TSE5      ;OLD LINE DISPLACEMENT
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
    35472  35022          LDA     3,DSC.,2 ;NOW ADJUST DATA POINTERS
    35473 166032          SGE     3,1      ;WAS DATA AT (DSC) AFFECTED ?
    35474    461          JMP     EDML6    ;   NO
    35475 166414          SEQ     3,1      ;WAS LINE AT (DSC) CHANGED ?
    35476    447          JMP     EDML5    ;   NO
    35477  20334          LDA     0,TSE1   ;   YES
    35500 100512          NEGL#   0,0,SZC  ;WAS A NEW LINE INSERTED ?
    35501    444          JMP     EDML5    ;   YES
    35502 126400          SUB     1,1      ;   NO, DATA WAS REPLACED
    35503  45020          STA     1,DWC.,2 ;ASSUME NO DATA STATEMENTS
    35504 157000 EDML3:   ADD     2,3      ;GET DSC ABSOLUTE POINTER
    35505  35401          LDA     3,1,3    ;STATEMENT RELATIVE POSITION
    35506 165400          INC     3,1      ;ASSUME DATA STATEMENT
    35507  45020          STA     1,DWC.,2 ;DATA RELATIVE LOCATION
    35510 157000          ADD     2,3      ;ABSOLUTE STATEMENT LOCATION
    35511  25400          LDA     1,0,3    ;FIRST WORD OF STATEMENT
    35512  20064          LDA     0,C377
    35513 123400          AND     1,0
    35514 106700          SUBS    0,1      ;STATEMENT TYPE BYTE
    35515  34223          LDA     3,C125
    35516 136415          SNE     1,3      ;"DATA" STATEMENT ?
    35517    410          JMP     EDML4    ;   YES
    35520  11022          ISZ     DSC.,2   ;   NO, STEP DSC
    35521  11022          ISZ     DSC.,2
    35522  25001          LDA     1,VDT.,2
    35523  35022          LDA     3,DSC.,2
    35524 166033          SLS     3,1      ;END OF SLT ?
    35525    420          JMP     EDML5    ;   YES, NO DATA STATEMENT
    35526    756          JMP     EDML3    ;   NO, CONTINUE SCAN

    35527  24062 EDML4:   LDA     1,C300   ;DATA STATEMENT FOUND
    35530 107400          AND     0,1      ;NUMBER TYPE * 2^6
    35531 122400          SUB     1,0      ;NUMBER OF DATA ELEMENTS
    35532  41021          STA     0,DEC.,2 ;SET DATA ELEMENT COUNTER
    35533 127120          ADDZL   1,1
    35534 127120          ADDZL   1,1
    35535 125300          MOVS    1,1      ;NUMBER TYPE * 2^2
    35536  21024          LDA     0,FLAG.,2
    35537  34402          LDA     3,.+2
    35540 163401          AND     3,0,SKP
    35541 177763          177763
    35542 123000          ADD     1,0      ;NEW NUMBER TYPE IN FLAG WORD
    35543  41024          STA     0,FLAG.,2
    35544    411          JMP     EDML6
```

```
;                    << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
35545  24334 EDML5: LDA     1,TSE1     ;LINE AT (DSC) WAS MOVED
35546 107000        ADD     0,1
35547  45022        STA     1,DSC.,2
35550  21020        LDA     0,DWC.,2
35551  24342        LDA     1,TSE7
35552 107000        ADD     0,1
35553 101014        SKZ     0,0
35554  45020        STA     1,DWC.,2
35555  35014 EDML6: LDA     3,FNS.,2   ;PREPARE TO ADJUST FOR-NEXT STACK
35556  25016        LDA     1,GSS.,2
35557 157000        ADD     2,3        ;.FNS
35560 133000        ADD     1,2        ;.GSS
35561 172033        SLS     3,2        ;END OF FOR-NEXT STACK ?
35562    413        JMP     EDML7      ;   YES
35563  21401        LDA     0,1,3      ;   NO
35564  24340        LDA     1,TSE5
35565 106032        SGE     0,1        ;WAS THIS ONE AFFECTED ?
35566    404        JMP     .+4        ;   NO
35567  24342        LDA     1,TSE7     ;   YES
35570 123000        ADD     1,0        ;ADD DISPLACEMENT
35571  41401        STA     0,1,3
35572  20026        LDA     0,C6
35573 117000        ADD     0,3        ;STEP STACK POINTER
35574    765        JMP     EDML6+4

35575  34355 EDML7: LDA     3,.BUS     ;PREPARE TO ADJUST GOSUB STACK
35576  21461        LDA     0,XOB.,3
35577 117000        ADD     0,3        ;.SLT
35600 156033        SLS     2,3        ;END OF GOSUB STACK ?
35601    412        JMP     EDML8      ;   YES
35602  21000        LDA     0,0,2      ;   NO
35603  24340        LDA     1,TSE5
35604 106032        SGE     0,1        ;WAS THIS ONE AFFECTED ?
35605    404        JMP     .+4        ;   NO
35606  24342        LDA     1,TSE7     ;   YES
35607 123000        ADD     1,0        ;ADD DISPLACEMENT
35610  41000        STA     0,0,2
35611 151400        INC     2,2        ;STEP STACK POINTER
35612    766        JMP     EDML7+3

35613  30355 EDML8: LDA     2,.BUS
35614   6102        FLAGCHECK          ;DELETE COMMAND ?
35615  20024        FLAG.+SKIP0
35616   2000        2000
35617    403        JMP     EDM8A      ;   NO
35620   2401        JMP     @.+1       ;   YES
35621  33443        DLTE1
35622  20334 EDM8A: LDA     0,TSE1
35623 101112        SSP     0,0        ;DELETE A LINE ?
35624   2306        JMP     @.EDUN     ;   YES
35625  20346        LDA     0,LNO      ;   NO
35626 101015        SNZ     0,0        ;EXECUTE MODE ?
35627    402        JMP     EDML9      ;   YES
35630   2306        JMP     @.EDUN     ;   NO
```

```
;                          << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062!  >>
    35631   34355 EDML9: LDA     3,.BUS
    35632   25424        LDA     1,FLAG.,3
    35633   20273        LDA     0,MFLAG
    35634  107400        AND     0,1
    35635  106000        ADC     0,1
    35636   45424        STA     1,FLAG.,3
    35637   21405        LDA     0,SLT.,3
    35640   41402        STA     0,PLC.,3
    35641   30005        LDA     2,RUP      ;SET UP FOR "RUN"
    35642   21002        LDA     0,FBA.,2
    35643   41005        STA     0,OBP.,2
    35644   31025        LDA     2,DFT.,2
    35645   55371        STA     3,FDA+CHM1,2
    35646   34402        LDA     3,.+2
    35647    2117        JMP     @BUMPU&377
    35650  100373        @373
    35651   30074 CRTL:  LDA     2,ETSF     ;CHECK RUN TIME LIMIT
    35652  151014        SKZ     2,2        ;TIME SLICE FLAG SET ?
    35653    2117        JMP     @BUMPU&377;   YES
    35654    1400        JMP     0,3        ;   NO



;  THIS ROUTINE TURNS ON THE MSB OF THE LINE #,  THE
;ADDRESS OF WHICH MUST BE IN A2.

    35655   50342 LMAP:  STA     2,TSE7
    35656   21000        LDA     0,0,2      ;LINE#
    35657  101100        MOVL    0,0
    35660  101240        MOVOR   0,0        ;TURN ON MSB
    35661   41000        STA     0,0,2
    35662   30355        LDA     2,.BUS
    35663    1400        JMP     0,3
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;   THIS SECTION RENUMBERS THE PROGRAM INTO THE UNLISTABLE ZONE
 35664   6277 RNUM:  JSR     @.CPYP
 35665  30355 RNUM2: LDA     2,.BUS    ;START SCAN OF SLT
 35666 126400         SUB     1,1
 35667  45055         STA     1,PSTS.,2;RESET STATUS
 35670  25005         LDA     1,SLT.,2 ;   TO CHANGE LINE NUMBERS
 35671  45002         STA     1,PLC.,2
 35672  35002 RNUM1: LDA     3,PLC.,2
 35673  25001         LDA     1,VDT.,2
 35674 166033         SLS     3,1       ;END OF SLT ?
 35675    411         JMP     RNUM3     ;   YES
 35676 173000         ADD     3,2       ;   NO, COMPUTE ABS ADDR OF LINE #
 35677   4756         JSR     LMAP      ;CONVERT LINE# TO BINARY
 35700  35055         LDA     3,PSTS.,2
 35701 117000         ADD     0,3       ;ACCUMULATE STATUS
 35702  55055         STA     3,PSTS.,2
 35703  11002         ISZ     PLC.,2    ;STEP TO NEXT LINE
 35704  11002         ISZ     PLC.,2
 35705    765         JMP     RNUM1

 35706 126400 RNUM3: SUB     1,1       ;RESCAN TO CHANGE LNO REFERENCES
 35707  45075         STA     1,TSU.+2,2
 35710  25005         LDA     1,SLT.,2
 35711  45002         STA     1,PLC.,2
 35712   4472 RNUM4: JSR     FLNR      ;FIND NEXT LINE NO. REFERENCE
 35713    413         JMP     POKE      ;   NONE FOUND, DONE
 35714   4741         JSR     LMAP      ;CONV. LINE#
 35715  25055         LDA     1,PSTS.,2
 35716 124700         NEGS    1,1       ;SWAPPED STATUS
 35717  22342         LDA     0,@TSE7   ;BINARY LINE# + 100000
 35720 131000         MOV     1,2       ;XOR A0 & A1 INTO A1
 35721 113520         ANDZL   0,2
 35722 107000         ADD     0,1
 35723 146400         SUB     2,1
 35724  46342         STA     1,@TSE7   ;REF LINE# XOR'ED WITH SWAPPED STATUS
 35725    765         JMP     RNUM4

 35726  30355 POKE:  LDA     2,.BUS
 35727  35005         LDA     3,SLT.,2
 35730 157000         ADD     2,3
 35731  25401         LDA     1,1,3     ;2ND WORD OF SLT
 35732  20036         LDA     0,C16
 35733 107400         AND     0,1       ;ISOLATE 3 LSB'S OF SLT WORD
 35734 122245         ADCOR   1,0,SNR   ;LSB'S = 110X ?
 35735 105000         MOV     0,1       ;   YES, USE 111X
 35736 125220         MOVZR   1,1       ;REMOVE L.O. BIT
 35737 135300         MOVS    1,3       ;A3 HAS LIST PROTECT BIT 9
 35740 127100         ADDL    1,1
 35741 127100         ADDL    1,1
 35742 125100         MOVL    1,1
 35743 137000         ADD     1,3       ;A3 HAS LP BITS 9 & 5
 35744 125300         MOVS    1,1
 35745 137000         ADD     1,3       ;A3 HAS LP BITS 15,9, & 5
 35746  20433         LDA     0,LPMSK
 35747 117400         AND     0,3       ;A3 = LIST PROTECT BITS ONLY
 35750  21024         LDA     0,FLAG.,2
 35751  24431         LDA     1,MLPM
 35752 123400         AND     1,0       ;TURN OFF LP BITS IN FLAG WORD
```

```
35753 163000          ADD     3,0         ;SET LP BITS IN RFW
35754  40424          STA     0,RFW       ;REAL FLAG WORD
35755  25055          LDA     1,PSTS.,2
35756  34425          LDA     3,PMSK      ;LP BITS IN STATUS
35757 167400          AND     3,1
35760 135000          MOV     1,3         ;XOR STATUS & RFW
35761 117520          ANDZL   0,3
35762 107000          ADD     0,1
35763 166400          SUB     3,1         ;A1 = FLAG WORD
35764  45024          STA     1,FLAG.,2
35765  25054          LDA     1,FL1.,2    ;GENERATE FL1
35766 103100          ADDL    0,0         ;ROTATE RFW LEFT 2 BITS
35767 125200          MOVR    1,1         ;SET CARRY = NO-INITIALIZE BIT
35770 101100          MOVL    0,0         ;RETAIN NO-INITIALIZE BIT
35771  41054          STA     0,FL1.,2
35772  34100          LDA     3,.INFO     ;GENERATE NEW FL2
35773  25440          LDA     1,HRS.,3
35774 124120          COMZL   1,1
35775  45056          STA     1,FL2.,2
35776   2401          JMP     @POKEY
35777  34051 POKEY:   RUNC1
36000      0 RFW:     0           ;REAL FLAG WORD
36001 101040 LPMSK:   101040      ;LIST PROTECT MASK BITS
36002  76737 MLPM:    76737
36003 111240 PMSK:    111240      ;LIST + OPTION PROTECT BITS
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
    36004   54333 FLNR:  STA     3,TSE      ;FIND LINE NUMBER REFERENCES
    36005   34355         LDA     3,.BUS
    36006   54040         STA     3,SBA      ;FORCE REFERENCES TO PROG AREA
    36007   15475         DSZ     TSU.2,3    ;Decrement count of line nbrs in current stmt
    36010  100010         NOP
    36011   31475         LDA     2,TSU.2,3  ;Load updated count
    36012  150015         COM#    2,2,SNR    ;Initial entry to FLNR?
    36013     422         JMP     FLNFR      ;  Yes, Advance to first line
    36014  151014         SKZ     2,2        ;End of current GOTO/GOSUB?
    36015     414         JMP     FLNNL      ;  No, Return next line number
    36016    6422 FLNNS: JSR     @.2NXSTM   ;Get next statement
    36017    2333           JMP   @TSE       ;  End of program, use nonskip return
    36020   24051 FLNCS: LDA     1,C100
    36021  122654         SUBOR#  1,0,SZR    ;GOTO or GOSUB?
    36022     774         JMP     FLNNS      ;  No, Try next statement
    36023   21000         LDA     0,0,2      ;Load first word of statement
    36024   24064         LDA     1,C377
    36025  123400         AND     1,0        ;Extract number of line#s (may be ON x GOTO/GOSUB)
    36026   34355         LDA     3,.BUS     ;Load partition address
    36027   41475         STA     0,TSU.2,3  ;NUMBER OF LINE NUMBERS
    36030   51476         STA     2,TSU.3,3  ;Set line number pointer
    36031   11476 FLNNL: ISZ     TSU.3,3    ;Increment pointer to next line number
    36032   31476         LDA     2,TSU.3,3  ;Return address of line# in A2
    36033   10333         ISZ     TSE
    36034    2333         JMP     @TSE       ;SKIP RETURN

    36035    6404 FLNFR: JSR     @.FRSTM    ;Initialize NXSTM and get first stmt
    36036    2333           JMP   @TSE       ;  End of pgm, Use nonskip return
    36037     761         JMP     FLNCS      ;Check if GOTO/GOSUB

    36040   43432 .2NXSTM:  NXSTM
    36041   43416 .FRSTM:  FRSTM
```

```
;                    << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>

36042  54333 WORD:  STA     3,TSE     ;OUTPUT WORD FROM TABLE
36043  50335         STA     2,TSE2
36044  30056         LDA     2,C240
36045   6132         OUTBYTE
36046  30335         LDA     2,TSE2
36047  34327         LDA     3,.WTBL
36050 173000         ADD     3,2
36051  21000         LDA     0,0,2
36052  25114         LDA     1,WTL,2
36053 111123         MOVZL   0,2,SNC
36054 125121         MOVZL   1,1,SKP
36055 141221         MOVZR   2,0,SKP
36056 125221         MOVZR   1,1,SKP
36057 126400 WORD1:  SUB     1,1
36060  44334         STA     1,TSE1
36061 101120         MOVZL   0,0
36062  30026 WORD2:  LDA     2,C6
36063  50335         STA     2,TSE2
36064  14335         DSZ     TSE2
36065 101121         MOVZL   0,0,SKP
36066    403         JMP     .+3
36067 151100         MOVL    2,2
36070    774         JMP     .-4
36071  40335         STA     0,TSE2
36072   6132         OUTBYTE
36073  20335         LDA     0,TSE2
36074 101014         SKZ     0,0
36075    765         JMP     WORD2
36076  20334         LDA     0,TSE1
36077 111005         MOV     0,2,SNR
36100    407         JMP     .+7
36101  24245         LDA     1,C333
36102 106033         SLS     0,1
36103    754         JMP     WORD1
36104  20062         LDA     0,C300
36105 113000         ADD     0,2
36106   6132         OUTBYTE
36107  30056         LDA     2,C240
36110  24337         LDA     1,TSE4
36111  20223         LDA     0,C125
36112 106014         ADC#    0,1,SZR   ;"REM" STATEMENT ?
36113   6132         OUTBYTE           ;  NO, STORE TRAILING SPACE
36114   2333         JMP     @TSE
```

```
;                          << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE *********
    36115     373 EDTBL: 373      ;  *    OPERATOR CODE TABLE FOR EDEX
    36116     371        371      ;  +
    36117     361        361      ;  ,
    36120     370        370      ;  -
    36121     350        350      ;  .
    36122     372        372      ;  /


    36123     256 LSTBL: 256      ;  .    350   OPERATOR CODE TABLE FOR LISTER
    36124     242        242      ;  "    351
    36125  137274        137274   ;  <>   352
    36126     276        276      ;  >    353
    36127  136676        136676   ;  >=   354
    36130     275        275      ;  =    355
    36131  136674        136674   ;  <=   356
    36132     274        274      ;  <    357
    36133     273        273      ;  ;    360
    36134     254        254      ;  ,    361
    36135     250        250      ;  (    362
    36136     333        333      ;  [    363
    36137     254        254      ;  ,    364
    36140     300        300      ;  @    365
    36141     251        251      ;  )    366
    36142     335        335      ;  ]    367
    36143     255        255      ;  -    370
    36144     253        253      ;  +    371
    36145     257        257      ;  /    372
    36146     252        252      ;  *    373
    36147     336        336      ;  ^    374
    36150     300        300
    36151     272        272      ;  :    376
    36152     243        243      ;  #    377



    36153      27        27
    36154   20002 LWRRL: LDA     0,C2       ;ERROR IN WRITE ITEM
    36155  152400        SUB     2,2
    36156  162415        SNE     3,0
    36157    2412        JMP     @.LW5      ;CHANNEL NOT OPEN
    36160   20773        LDA     0,LWRRL-1
    36161  162414        SEQ     3,0
    36162    2317        JMP     @.PERM
    36163   20003        LDA     0,C3       ;RECORD IS LOCKED
    36164    6101        CALL
    36165      57        SIGPAUSE
    36166    6117        BUMPUSER
    36167  126000        ADC     1,1
    36170    2402        JMP     @.LW


    36171   36536 .LW5:  LWRTE+5
    36172   36546 .LW:   LWRTC

                .EOT   ;  "PROTECT" R9.0 SOURCE #4
```

```
;                    << SI = R94PROTECTSC;  BD = 1/A.SPP4.9062! >>

    36173  36173 RELO1:.
           35777        .LOC   POKEY
    35777  36203        CKEY
           36173        .LOC   RELO1

;-----------------------------------------------------------------
;
; THIS ROUTINE ADDS THE LEVEL 1 PICO-N PROTECT KEY TO A
;CUSTOMER APPLICATION PROGRAM.   THE OPERATOR ENTERS A
;LEVEL 1 KEY FOLLOWING THE PROGRAM NAME, I.E.
;    PROTECT  PROGRAMNAME,KEY   -OR-   PROTECT  ,KEY
;THE KEY ENTERED IS IN HEX.
;THE KEY IS MASSAGED DIFFERENTLY BEFORE BEING MOVED TO THE
;CUSTOMER PROGRAM.
;
;-----------------------------------------------------------------

    36173      0 .MT:   0        ;A(CONVERSION TBL)
    36174      0 .MIDX: 0
    36175  36176 .MPYR: .+1
    36176      1 MPYR:  1        ;CUST KEY CONVERSION TBL
    36177     20        20
    36200    400        400
    36201  10000        10000
    36202      0 BI:    0                  ;BYTE INDEX

    36203  20447 CKEY:  LDA     0,OKEY     ;see if we have a key yet
    36204 101414         INC#    0,0,SZR   ;-1 indicates that we don't yet ??
    36205    405         JMP     CKEY1     ;   YES, we already know it
    36206   4457         JSR     SKEY      ;SCAN INPUT BUFFER TO KEY
    36207   4512         JSR     SKEY1     ;SCAN KEY
    36210  30765         LDA     2,.MPYR
    36211   4527         JSR     CALCK     ;CONVERT TO BINARY
    36212  20440 CKEY1:  LDA     0,OKEY    ;BINARY KEY
    36213   6421         JSR     @.ENC     ;GENERATE 8 WORDS + CHECKSUM
    36214  34355         LDA     3,.BUS
    36215  21404         LDA     0,UPS.,3
    36216 117000         ADD     0,3
    36217  21010         LDA     0,10,2    ;CHECKSUM
    36220  41776         STA     0,-2,3    ;SAVE IN UPS-2
    36221 141000         MOV     2,0       ;A(1ST SOURCE WORD)
    36222  30355         LDA     2,.BUS
    36223  25005         LDA     1,SLT.,2
    36224 133000         ADD     1,2
    36225  24030         LDA     1,C10
    36226 132400         SUB     1,2       ;A(DESTINATION)
    36227  24027         LDA     1,C7
    36230 107000         ADD     0,1       ;A(LAST SOURCE WORD)
    36231   6101         CALL               ;MOVE 8 WORDS TO USER AREA
    36232 100015         MOVEWORDS
    36233    457         JMP     RIBP      ;RESTORE IBP & CONTINUE
```

```
;                       << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>

 36234  36430 .ENC:  ENCDE

 36235   6133 BADK:  OUTTEXT
 36236 106677        .TXTF   ;<215>?
 36237 120311  I
 36240 147326 NV
 36241 140714 AL
 36242 144704 ID
 36243 120313  K
 36244 142731 EY
 36245 103400 <207>;

 36246   6141        STOUTPUT
 36247   6101        CALL
 36250 100000        SCOPE
```

```
;                       << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>

;---------------------------------------------------------------
;
;THIS SUBROUTINE SCANS PAST THE FILENAME IN THE IOB &
;CHANGES THE COMMA TO A CARRIAGE RETURN.
;
;---------------------------------------------------------------

    36251       0 RKEY:  0                    ;RETURN ADDRESS FROM SKEY & CALCK
    36252  177777 OKEY:  177777 ;Binary key, none known = -1
                        ;  since this processor does not swap and
                        ;  is replaced by SAVE before completion
                        ;  the value can be assumed to be initialized
                        ;  by the LOADING of the processor
                        ;  If the other user does get in before SAVE the
                        ;  current processor in REX has been cleared
                        ;  by SWPO so a new version of PROTECT will be
                        ;  loaded anyway.
    36253   36254 .CUK:  .+1
    36254       5 CUK:   .BLK   5             ;\215\ + KEY BYTES
    36261  177777        -1                   ;TERMINATOR
    36262       0 KEYC:  0                    ;COUNT OF # OF KEY BYTES
    36263   34051 RC1:   RUNC1
    36264       0 SIBP:  0                    ;OLD IBP

    36265   54764 SKEY:  STA     3,RKEY
    36266   30005        LDA     2,RUP
    36267   25004        LDA     1,IBP.,2
    36270   44774        STA     1,SIBP       ;SAVE OLD IBP

;SCAN PAST FILENAME.................

    36271  102000        ADC     0,0
    36272    6125        INBYTE               ;IGNORE BLANKS
    36273   20055        LDA     0,C215
    36274  142415        SNE     2,0          ;FILENAME GIVEN ?
    36275     415        JMP     RIBP         ;   NO
    36276   20236        LDA     0,C254       ;   MAYBE...
    36277  142415        SNE     2,0          ;COMMA ?
    36300     416        JMP     PKEY         ;   YES, PROCESS KEY
    36301  102000 SFN:   ADC     0,0          ;   NO
    36302    6126        INSTBYTE             ;SCAN FILENAME
    36303   20055        LDA     0,C215
    36304  142415        SNE     2,0          ;END OF FILENAME & NO KEY ?
    36305     405        JMP     RIBP         ;   YES
    36306   20236        LDA     0,C254       ;COMMA
    36307  142415        SNE     2,0          ;END OF FILENAME & KEY ?
    36310     406        JMP     PKEY         ;   YES
    36311     770        JMP     SFN          ;   NO

    36312   30005 RIBP:  LDA     2,RUP        ;RESTORE IBP & RETURN
    36313   24751        LDA     1,SIBP
    36314   45004        STA     1,IBP.,2
    36315    2746        JMP     @RC1

    36316   20055 PKEY:  LDA     0,C215       ;CHANGE COMMA TO '215' IN INPUT BUFFER
    36317    6134        PUTBYTE              ;A1 = B(',' IN THE INPUT BUFFER)
    36320    2731        JMP     @RKEY        ;RETURN
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;-------------------------------------------------------------------
;
;THIS SUBROUTINE READS THE KEY FROM THE INPUT BUFFER & SAVES
;IT IN 'CUK'
;INPUT:  IOB
;OUTPUT:   'CUK' CONTAINS:    C215,LSB,...,MSB
;
;-------------------------------------------------------------------

    36321   54730 SKEY1: STA     3,RKEY
    36322   20024        LDA     0,C4
    36323   40737        STA     0,KEYC
    36324    6126 RDKY:  INSTBYTE
    36325   24735        LDA     1,KEYC
    36326   34725        LDA     3,.CUK
    36327  137000        ADD     1,3
    36330   51400        STA     2,0,3      ;SAVE BYTE
    36331   20055        LDA     0,C215
    36332  142415        SNE     2,0        ;END OF KEY
    36333    2716        JMP     @RKEY      ;   YES, RETURN
    36334   14726        DSZ     KEYC       ;END OF KEY ?
    36335     767        JMP     RDKY       ;   NO
    36336   41777        STA     0,-1,3     ;   YES, SAVE C215
    36337    2712        JMP     @RKEY      ;RETURN
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>

;-----------------------------------------------------------------
;
;THIS SUBROUTINE CONVERTS THE INPUT KEY TO BINARY.
;INPUT:    A2 = A(CONVERSION TABLE)
;          A3 = RETURN
;          'CUK' = ASCII KEY CHARACTERS
;OUTPUT:   'OKEY' = BINARY KEY
;
;-----------------------------------------------------------------

    36340   54711 CALCK: STA     3,RKEY
    36341   50632         STA     2,.MT
    36342  126400         SUB     1,1
    36343   44707         STA     1,OKEY    ;INITIALIZE BINARY KEY
    36344   34707         LDA     3,.CUK
    36345   31400         LDA     2,0,3     ;SCAN TO 1ST BYTE
    36346  175400         INC     3,3
    36347  142414         SEQ     2,0       ;C215 ?
    36350     775         JMP     .-3       ;   NO
    36351   54631         STA     3,BI
    36352   30621         LDA     2,.MT     ;   YES
    36353   50621         STA     2,.MIDX
    36354   34626 CBO:    LDA     3,BI
    36355   21400         LDA     0,0,3     ;GET INPUT BYTE
    36356  101415         INC#    0,0,SNR   ;DONE ?
    36357    2672         JMP     @RKEY     ;   YES
    36360   24052         LDA     1,C177    ;   NO
    36361  123400         AND     1,0
    36362   24214         LDA     1,C71
    36363  106433         SLE     0,1       ;0 TO 9 ?
    36364     416         JMP     AF        ;   NO
    36365   24213         LDA     1,C60     ;   MAYBE...
    36366  106032         SGE     0,1
    36367     646         JMP     BADK      ;BAD KEY
    36370  122400 CO:     SUB     1,0       ;HEX 0 TO 9, CONVERT TO BINARY
    36371   30603         LDA     2,.MIDX
    36372   31000         LDA     2,0,2     ;MULTIPLIER
    36373   10601         ISZ     .MIDX
    36374    6116         BINMULTIPLY       ;(A0)(A2) = A3,A0
    36375   24655         LDA     1,OKEY
    36376  107000         ADD     0,1
    36377   44653         STA     1,OKEY
    36400   10602         ISZ     BI
    36401     753         JMP     CBO

    36402   24051 AF:     LDA     1,C100
    36403  106432         SGR     0,1       ;A THRU F ?
    36404     631         JMP     BADK      ;   NO, INVALID KEY
    36405   24405         LDA     1,C106    ;   MAYBE...
    36406  106433         SLE     0,1
    36407     626         JMP     BADK
    36410   24403         LDA     1,C67     ;   YES
    36411     757         JMP     CO

    36412     106 C106:   106
    36413      67 C67:    67
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>

;---------------------------------------------------------------

;GENERATE 8 WORDS + CHECKSUM TO BE ENCODED INTO THE
;BASIC PROGRAM AS THE CUSTOMER PROTECT MECHANISM.
;
;INPUT:
;  A0 = LEVEL 1 KEY
;  A3 = RETURN
;
;OUTPUT:
;  A2 = ADDRESS OF THE FOLLOWING 9 WORD TABLE:
;       WORD 0  :KEY LOCATION + PROTECT INDICATOR:
;                BIT 0-2 = DISPLACEMENT OF THE KEY WITHIN
;                          THE NEXT 7 WORDS.
;                  "   3  = 0 - USE BIT 4 AS PROTECT INDICATOR
;                           1 -  "   "  5  "      "        "
;                  "  4/5 = 0 - PROGRAM IS NOT PROTECTED
;                           1 -     "       IS          "
;                  "  6-15= RANDOM
;       WORD 1-7: SIX RANDOM #'S & ONE LEVEL 1 KEY.  THE KEY IS
;                 SWAPPED & NEGATED.
;       WORD 8  : CHECKSUM OF WORDS 0-7
;
;---------------------------------------------------------------
```

```
;                    << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>

 36414  36415 .ENCR:ENCR
 36415     11 ENCR:  .BLK    11          ;RESULTS
 36426      0 L1KEY: 0        ;LEVEL 1 KEY
 36427      0 RTN:   0
 36430  54777 ENCDE: STA     3,RTN
 36431  40775         STA     0,L1KEY
 36432   4451         JSR     RNDM        ;GENERATE RANDOM # (IN A0)
 36433  24030         LDA     1,C10       ;BIT 3 MASK
 36434  30042         LDA     2,C20       ;BIT 4 MASK
 36435 107404         AND     0,1,SZR     ;BIT 3 ON ?
 36436 151120         MOVZL   2,2         ;   YES, USE BIT 5 AS PROTECT INDICATOR
 36437 150000         COM     2,2         ;TURN OFF BIT 4 OR 5
 36440 143400         AND     2,0
 36441  40754         STA     0,ENCR      ;WORD 0
 36442  24437         LDA     1,CM7
 36443  30751         LDA     2,.ENCR
 36444   4437         JSR     RNDM
 36445  41001         STA     0,1,2       ;WORDS 1-7
 36446 151400         INC     2,2
 36447 125404         INC     1,1,SZR     ;DONE ?
 36450    774         JMP     .-4         ;   NO
 36451  20744         LDA     0,ENCR
 36452  30742         LDA     2,.ENCR     ;A(TABLE)
 36453  24027         LDA     1,C7
 36454 123400         AND     1,0         ;CALCULATE KEY LOCATION
 36455 122414         SEQ     1,0         ;0-6 MAX
 36456    403         JMP     ENC1
 36457  14736         DSZ     ENCR        ;USE 6 FOR 7
 36460  20026         LDA     0,C6
 36461 113000 ENC1:   ADD     0,2
 36462  20744         LDA     0,L1KEY     ;LEVEL 1 KEY
 36463 101300         MOVS    0,0
 36464 100400         NEG     0,0
 36465  41001         STA     0,1,2       ;STORE IN TABLE
 36466 102440         SUBO    0,0         ;NOW CALC CHECKSUM
 36467  34413         LDA     3,CM10
 36470  30724         LDA     2,.ENCR
 36471  25000         LDA     1,0,2
 36472 123200         ADDR    1,0
 36473 151400         INC     2,2
 36474 175404         INC     3,3,SZR
 36475    774         JMP     .-4
 36476  41000         STA     0,0,2       ;SAVE CHECKSUM IN WORD 8
 36477  30715         LDA     2,.ENCR
 36500   2727         JMP     @RTN        ;RETURN
 36501 177771 CM7:    -7
 36502 177770 CM10:   -10
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;-------------------------------------------------------------
;
;GENERATE A RANDOM #
;
;INPUT:   A3 = RETURN
;
;OUTPUT:
;   A0 = RANDOM #
;   A1, A2 = UNCHANGED
;
;-------------------------------------------------------------

    36503   54425 RNDM:  STA     3, RTN1
    36504   44422         STA     1, SA1
    36505   50422         STA     2, SA2
    36506   34100         LDA     3, .INFO
    36507   21441         LDA     0, TSC., 3 ;TEN SECOND CLOCK
    36510   24037         LDA     1, C17
    36511   30414         LDA     2, RNR
    36512  143000         ADD     2, 0
    36513  111000         MOV     0, 2
    36514  133400         AND     1, 2
    36515  150400         NEG     2, 2
    36516  101100         MOVL    0, 0        ;SHIFT 0 TO 16 PLACES
    36517  151404         INC     2, 2, SZR
    36520     776         JMP     .-2
    36521   40404         STA     0, RNR
    36522   24404         LDA     1, SA1      ;A0 = RANDOM #
    36523   30404         LDA     2, SA2
    36524    2404         JMP     @RTN1

    36525       0 RNR:   0
    36526       0 SA1:   0
    36527       0 SA2:   0
    36530       0 RTN1:  0
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>

;  "BUSINESS PROTECT" SOURCE #5 OF 6 TAPES FOR "IRIS" R8.0
;********* THIS PAGE IS GARBAGE *******

    36531   30355 LWRTE: LDA     2,.BUS    ;WRITE ONE LINE OF LISTING
    36532   55076        STA     3,TSU.+3,2
    36533   21403        LDA     0,LBA.,3
    36534   24222        LDA     1,C121    ;80 CHARACTER LINE
    36535  122400        SUB     1,0
    36536   25405        LDA     1,OBP.,3
    36537  106433        SLE     0,1       ;ROOM FOR ANOTHER LINE ?
    36540     450        JMP     LWRTX     ;   YES
    36541    6132        OUTBYTE           ;   NO
    36542    6141        STOUTPUT
    36543    6101        CALL
    36544  100011        WONA
    36545     443        JMP     LWRTX
    36546  102000 LWRTC: ADC     0,0       ;CHANNEL LIST
    36547   40447        STA     0,LWICB
    36550   34005        LDA     3,RUP
    36551  125014        SKZ     1,1       ;PUNCHING LEADER ?
    36552     412        JMP     LRTC1     ;   YES
    36553   21405        LDA     0,OBP.,3  ;   NO
    36554   31403        LDA     2,LBA.,3
    36555  112033        SLS     0,2       ;BUFFER FULL ?
    36556   15405        DSZ     OBP.,3    ;   YES
    36557   34355        LDA     3,.BUS
    36560   35473        LDA     3,TSU.,3
    36561   30055        LDA     2,C215
    36562  174137        COMZL#  3,3,SBN   ;TAPE MODE ?
    36563    6132        OUTBYTE           ;   NO, STORE A RETURN
    36564   34005 LRTC1: LDA     3,RUP
    36565   21402        LDA     0,FBA.,3
    36566  101400        INC     0,0
    36567   25405        LDA     1,OBP.,3
    36570  101400        INC     0,0       ;IGNORE RETURN
    36571  106400        SUB     0,1
    36572  101400        INC     0,0       ;IGNORE NULL
    36573   40427        STA     0,LWICB+4
    36574   44425        STA     1,LWICB+3
    36575   30420        LDA     2,LWICB-1
    36576  102400        SUB     0,0
    36577   40420        STA     0,LWICB+1;RESET ITEM # TO ZERO
    36600   40041        STA     0,DBA
    36601    6106        CHANNEL
    36602   20034        WRITITEM
    36603    2411        JMP     @.LWR
    36604   34005        LDA     3,RUP
    36605   21402        LDA     0,FBA.,3
    36606   41405        STA     0,OBP.,3
    36607    6300        JSR     @.CRTL
    36610   22004 LWRTX: LDA     0,@PIB
    36611   40040        STA     0,SBA
    36612   34355        LDA     3,.BUS
    36613    3476        JMP     @TSU.+3,3
    36614   36154 .LWR:  LWRRL
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE *******
    36615   36616          LWICB   ;LIST ITEM CONTROL BLOCK
    36616 177777 LWICB:   -1      ;REC #
    36617      0          0       ;ITEM #

    36620  54333 OPLD:    STA     3,TSE     ;OPEN FOR "LOAD"
    36621   6277          JSR     @.CPYP
    36622   6125          INBYTE
    36623  20057          LDA     0,C244
    36624 142404          SUB     2,0,SZR   ;LOAD FROM A DEVICE ?
    36625 102000          ADC     0,0       ;   NO
    36626  40414 OPLDD:   STA     0,OPLDU   ;ENTRY TO DUMP TO A DEVICE
    36627 102400          SUB     0,0
    36630   6106          CHANNEL
    36631     27          CLEAR
    36632   6142          TRAPFAULT         ;CHANNEL #0 IS ILLEGAL !?
    36633 114010          30*K!NOP
    36634  34005          LDA     3,RUP
    36635  25404          LDA     1,IBP.,3
    36636 102400          SUB     0,0
    36637  40040          STA     0,SBA
    36640   4403          JSR     .+3
    36641     37          37
    36642 177777 OPLDU:   -1
    36643 171000          MOV     3,2       ;OPEN DEVICE OR FILE
    36644   6106          CHANNEL
    36645  40022          OPEN
    36646   2317          JMP     @.PERM
    36647  34005 OPLDX:   LDA     3,RUP
    36650  45404          STA     1,IBP.,3
    36651  15404          DSZ     IBP.,3
    36652  22004          LDA     0,@PIB
    36653  40041          STA     0,DBA
    36654   2333          JMP     @TSE

    36655  54333 BLDFL:   STA     3,TSE     ;BUILD FILE FOR "DUMP"
    36656   6125          INBYTE
    36657  20057          LDA     0,C244
    36660 142405          SUB     2,0,SNR   ;DUMP TO A DEVICE ?
    36661    745          JMP     OPLDD     ;   YES
    36662 102400          SUB     0,0       ;   NO
    36663  40040          STA     0,SBA
    36664   4407          JSR     .+7
    36665  77030          77030
    36666      1 BBCNT:   1
    36667 177777          -1
    36670      0          0
    36671 177777          -1
    36672      0          0
    36673 171000          MOV     3,2       ;BUILD A TEXT FILE
    36674   6106          CHANNEL
    36675     20          BUILD
    36676   2317          JMP     @.PERM
    36677    750          JMP     OPLDX
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062!  >>
;********** THIS PAGE IS GARBAGE **********
    36700  34355 LISTT: LDA     3,.BUS     ;TERMINATE LISTING
    36701  31473         LDA     2,TSU.,3
    36702 151015         SNZ     2,2        ;"LIST" MODE ?
    36703    412         JMP     LSTTO      ;   YES
    36704  30055         LDA     2,C215     ;   NO
    36705   6132         OUTBYTE
    36706   4414         JSR     LLDR       ;OUTPUT TRAILER IF "TAPE"
    36707 102400         SUB     0,0
    36710   6106         CHANNEL
    36711     26         CLOSE
    36712 175014         SKZ     3,3
    36713   2306         JMP     @.EDUN
    36714   6142         TRAPFAULT          ;CHANNEL #0 IS ILLEGAL !?

    36715   6132 LSTTO: OUTBYTE             ;OUTPUT ANYTHING IN BUFFER
    36716   6141         STOUTPUT
    36717   6101         CALL
    36720 100011         WONA
    36721   2306         JMP     @.EDUN


    36722  30355 LLDR:  LDA     2,.BUS     ;LEADER OR TRAILER
    36723  55075         STA     3,TSU.+2,2
    36724   6323         JSR     @.SNI
    36725  34355         LDA     3,.BUS
    36726  41477         STA     0,TSU.+4,3 ;SAVE END LINE #
    36727 102400         SUB     0,0
    36730   6101         CALL
    36731 100002         CHKCHANNEL
    36732    402         SKIP
    36733    414         JMP     LLDRX
    36734  20051         LDA     0,C100     ;CHANNEL NOT OPEN
    36735  40333         STA     0,TSE
    36736  30053         LDA     2,C200
    36737   6132         OUTBYTE
    36740  14333         DSZ     TSE
    36741    775         JMP     .-3
    36742 152400         SUB     2,2
    36743   6132         OUTBYTE
    36744   6141         STOUTPUT
    36745   6101         CALL
    36746 100011         WONA
    36747  34355 LLDRX: LDA     3,.BUS
    36750   3475         JMP     @TSU.+2,3


    36751  37513 .FSLN: FSLN
    36752 114631         114631
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE *******
    36753 102400 TAPE:  SUB     0,0         ;"TAPE" COMMAND
    36754   6275        JSR     @.CCAO      ;CLEAR CHANNEL #0
    36755 102521        SUBZL   0,0,SKP
    36756 102000 DUMP:  ADC     0,0         ;"DUMP" COMMAND
    36757  34355        LDA     3,.BUS
    36760  41473        STA     0,TSU.,3
    36761  20346        LDA     0,LNO
    36762  41474        STA     0,TSU.+1,3
    36763   6277        JSR     @.CPYP
    36764  34355        LDA     3,.BUS      ;CALCULATE PROGRAM SIZE
    36765  25406        LDA     1,UVS.,3
    36766  21404        LDA     0,UPS.,3
    36767 122432        SGR     1,0
    36770   6330        NCODERROR           ;NO PROGRAM EXISTS
    36771 134010        70*K!NOP
    36772  25401        LDA     1,VDT.,3
    36773  31405        LDA     2,SLT.,3
    36774 146640        SUBOR   2,1         ;# OF PROGRAM LINES
    36775  20212        LDA     0,C32       ;DIVIDE BY 26
    36776   6115        BINDIVIDE
    36777 175400        INC     3,3         ;ADD 1 BLOCK FOR HEADER
    37000  54666        STA     3,BBCNT
    37001  34355        LDA     3,.BUS
    37002  21473        LDA     0,TSU.,3
    37003 101415        INC#    0,0,SNR     ;"DUMP" ?
    37004   4651        JSR     BLDFL       ;  YES, BUILD THE FILE
    37005   4715        JSR     LLDR
    37006  34355        LDA     3,.BUS
    37007  21474        LDA     0,TSU.+1,3
    37010  40346        STA     0,LNO       ;STARTING LINE #
    37011  21477        LDA     0,TSU.+4,3; ENDING LINE #
    37012    407        JMP     LIST1

    37013  34355 LIST:  LDA     3,.BUS      ;"LIST" COMMAND
    37014 102400        SUB     0,0
    37015  41473        STA     0,TSU.,3
    37016   6275        JSR     @.CCAO      ;CLEAR CHANNEL ZERO
    37017   6277        JSR     @.CPYP
    37020   6323        JSR     @.SNI
    37021 101015 LIST1: SNZ     0,0         ;ENDING LINE # GIVEN ?
    37022  20730        LDA     0,.FSLN+1;  NO, USE 9999
    37023  34355        LDA     3,.BUS
    37024  41477        STA     0,TSU.+4,3
    37025 102400        SUB     0,0
    37026  41474        STA     0,TSU.+1,3
    37027  41475        STA     0,TSU.+2,3
    37030  31405        LDA     2,SLT.,3 ;ASSUME NO STARTING LINE #
    37031  20346        LDA     0,LNO
    37032 101015        SNZ     0,0         ;STARTING LINE # GIVEN ?
    37033    405        JMP     .+5         ;  NO
    37034   6715        JSR     @.FSLN      ;  YES, FIND IT
    37035    643        JMP     LISTT       ;LIST DONE, EXIT
    37036  34355        LDA     3,.BUS
    37037 172400        SUB     3,2
    37040  51402        STA     2,PLC.,3
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE *******
    37041  30355 DCOD:  LDA     2,.BUS     ;DECODE A STATEMENT
    37042  35002         LDA     3,PLC.,2
    37043  25001         LDA     1,VDT.,2
    37044 157000         ADD     2,3
    37045 147000         ADD     2,1
    37046  21400         LDA     0,0,3
    37047  31077         LDA     2,TSU.+4,2
    37050 112432         SGR     0,2        ;BEYOND LAST LINE TO LIST . . .
    37051 166033         SLS     3,1        ;OR END OF SLT ?
    37052    626         JMP     LISTT      ;   YES
    37053   6133 LISTL:  OUTTEXT            ;   NO
    37054 106600         215*K+200
    37055      0         0
    37056 102400         SUB     0,0        ;INITIALIZE FLAGS
    37057  40336         STA     0,TSE3
    37060  40341         STA     0,TSE6
    37061 102520         SUBZL   0,0
    37062  40344         STA     0,ONF
    37063  30355         LDA     2,.BUS
    37064  35002         LDA     3,PLC.,2   ;SET BYTE POINTER
    37065 157000         ADD     2,3
    37066  21401         LDA     0,1,3
    37067  24065         LDA     1,C400
    37070 123120         ADDZL   1,0
    37071  41003         STA     0,PBC.,2
    37072  11002         ISZ     PLC.,2     ;STEP LINE COUNTER
    37073  11002         ISZ     PLC.,2
    37074  25400         LDA     1,0,3      ;OUTPUT LINE NUMBER
    37075   6324         JSR     @.SNO
    37076   6331         NEXTBYTE           ;GET DIRECTIVE BYTE
    37077  50337         STA     2,TSE4
    37100  24223         LDA     1,C125
    37101  34355         LDA     3,.BUS
    37102  21475         LDA     0,TSU.+2,3
    37103 146004         ADC     2,1,SZR    ;"NEXT" STATEMENT ?
    37104    405         JMP     .+5        ;   NO
    37105 101014         SKZ     0,0        ;   YES
    37106 122000         ADC     1,0        ;DECREMENT INDENTATION
    37107  41475         STA     0,TSU.+2,3
    37110    403         JMP     .+3

    37111 125225         MOVZR   1,1,SNR    ;"FOR" STATEMENT ?
    37112  11475         ISZ     TSU.+2,3   ;   YES, INCREASE INDENTATION
    37113 101125         MOVZL   0,0,SNR    ;ANY INDENTATION ?
    37114    407         JMP     LSTL1      ;   NO
    37115  40340         STA     0,TSE5     ;   YES, SET COUNTER
    37116  30056         LDA     2,C240     ;STORE SPACES
    37117   6132         OUTBYTE
    37120  14340         DSZ     TSE5
    37121    775         JMP     .-3
    37122  30337         LDA     2,TSE4
    37123  24213 LSTL1:  LDA     1,C60
    37124 132400         SUB     1,2
    37125  34211         LDA     3,C22
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE ******
    37126  21403          LDA     0,PBC.,3
    37127 101220          MOVZR   0,0
    37130  36004          LDA     3,@PIB
    37131 117000          ADD     0,3
    37132  35400          LDA     3,0,3
    37133  20064          LDA     0,C377
    37134 163400          AND     3,0
    37135  40344          STA     0,ONF
    37136    537          JMP     LISTG+3

    37137   6326 LSTL2:   JSR     @.WORD    ;OUTPUT STATEMENT DIRECTIVE WORD
    37140  30337          LDA     2,TSE4
    37141  24223          LDA     1,C125
    37142 146415          SNE     2,1       ;"DATA" STATEMENT ?
    37143   2521          JMP     @.LSTD    ;   YES
    37144 132015          ADC#    1,2,SNR   ;"REM" STATEMENT ?
    37145    464          JMP     LISTR     ;   YES
    37146  24222          LDA     1,C121    ;   NO
    37147 102000          ADC     0,0
    37150 146015          ADC#    2,1,SNR   ;"DEF" STATEMENT ?
    37151  40341          STA     0,TSE6    ;   YES
    37152   6331 LISTX:   NEXTBYTE
    37153  20051          LDA     0,C100
    37154 142432          SGR     2,0       ;INTEGER CONSTANT ?
    37155   2511          JMP     @.LSTI    ;   YES
    37156  24224          LDA     1,C132    ;   NO
    37157 146432          SGR     2,1       ;USER FUNCTION ?
    37160   2510          JMP     @.LSTU    ;   YES
    37161  20226          LDA     0,C174    ;   NO
    37162 142032          SGE     2,0       ;PRIMITIVE FUNCTION ?
    37163   2502          JMP     @.LSTF    ;   YES
    37164  20053          LDA     0,C200    ;   NO
    37165 142032          SGE     2,0       ;FLOATING-POINT CONSTANT ?
    37166   2501          JMP     @.LSTK    ;   YES
    37167  20250          LDA     0,C340    ;   NO
    37170 142432          SGR     2,0       ;VARIABLE NUMBER ?
    37171    562          JMP     LISTZ     ;   YES
    37172  24270          LDA     1,C375    ;   NO
    37173 146415          SNE     2,1       ;SINGLE QUOTE ?
    37174   2302          JMP     @.DSQ     ;   YES
    37175  24251          LDA     1,C342    ;   NO
    37176 146415          SNE     2,1       ;END OF STATEMENT ?
    37177    530          JMP     LISTA     ;   YES
    37200 132014          ADC#    1,2,SZR   ;STATEMENT FOLLOWS ?
    37201    411          JMP     LSTX1     ;   NO
    37202  34355          LDA     3,.BUS    ;   YES, START NEW STATEMENT
    37203  31403          LDA     2,PBC.,3
    37204 151620          INCZR   2,2
    37205 151120          MOVZL   2,2
    37206  51403          STA     2,PBC.,3
    37207   6331          NEXTBYTE
    37210  50337          STA     2,TSE4
    37211    712          JMP     LSTL1
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE ********
    37212  24254 LSTX1: LDA     1,C350
    37213 146432         SGR     2,1         ;OPERATOR WORD ?
    37214   2455         JMP     @.LSTW      ;   YES
    37215  20437         LDA     0,LSTR1-1   ;   NO, MUST BE OPERATOR SYMBOL
    37216 113000         ADD     0,2
    37217  31000         LDA     2,0,2       ;LOOK UP OPERATOR SYMBOL
    37220  50334         STA     2,TSE1
    37221   6132         OUTBYTE
    37222  24334         LDA     1,TSE1
    37223 131300         MOVS    1,2
    37224 125112         SSP     1,1         ;MULTIPLE SYMBOL ?
    37225   6132         OUTBYTE             ;   YES, STORE 2ND SYMBOL
    37226  34231         LDA     3,C242
    37227 116404         SUB     0,3,SZR     ;QUOTE MARK ?
    37230    722         JMP     LISTX       ;   NO
    37231  54334 LISTR: STA     3,TSE1      ;BEGIN LITERAL STRING
    37232   6331         NEXTBYTE
    37233 151015         SNZ     2,2         ;TERMINATOR CODE ?
    37234    421         JMP     LSTR1       ;   YES
    37235  24334         LDA     1,TSE1      ;   NO
    37236 125014         SKZ     1,1         ;"REM" STATEMENT OR COMMENT ?
    37237    413         JMP     LSTRS       ;   YES
    37240  24056         LDA     1,C240      ;   NO
    37241  34064         LDA     3,C377
    37242 146033         SLS     2,1         ;CONTROL CHARACTER . . .
    37243 156415         SNE     2,3         ;OR RUBOUT ?
    37244  24231         LDA     1,C242      ;   NO
    37245 146414         SEQ     2,1         ;QUOTE SYMBOL ?
    37246    404         JMP     LSTRS       ;   NO
    37247  30233         LDA     2,C247      ;   YES
    37250   6132         OUTBYTE             ;STORE DOUBLE APOSTROPHE
    37251  30233         LDA     2,C247
    37252   6132 LSTRS: OUTBYTE
    37253    757         JMP     LISTR+1

    37254  35553         LSTBL-350
    37255  24334 LSTR1: LDA     1,TSE1
    37256 125014         SKZ     1,1         ;"REM" STATEMENT OR COMMENT ?
    37257    467         JMP     JLSTE       ;   YES
    37260  30231         LDA     2,C242      ;   NO, OUTPUT CLOSING QUOTE
    37261   6132 LISTS: OUTBYTE
    37262    670         JMP     LISTX


    37263  37602 .LSTC: LISTC
    37264  37432 .LSTD: LISTD
    37265  37574 .LSTF: LISTF
    37266  37571 .LSTI: LISTI
    37267  37545 .LSTK: LISTK
    37270  37532 .LSTU: LISTU
    37271  37543 .LSTW: LISTW
```

```
;                          << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE ********
    37272 146000 LISTG: ADC     2,1         ;"GOTO" OR "GOSUB" OPERATOR
    37273  30037        LDA     2,C17
    37274 132400        SUB     1,2
    37275   6326        JSR     @.WORD      ;<< ENTRY FROM LSTL1
    37276 111003        MOV     0,2,SNC     ;EXTRA SPACE NEEDED ?
    37277   6132        OUTBYTE             ;   YES
    37300  30355        LDA     2,.BUS
    37301  25003        LDA     1,PBC.,2
    37302 125620        INCZR   1,1         ;CHANGE TO WORD POINTER
    37303  36004        LDA     3,@PIB
    37304 167000        ADD     3,1
    37305  44333        STA     1,TSE
    37306  26333 LSTG1: LDA     1,@TSE      ;OUTPUT A LINE NUMBER
    37307   6324        JSR     @.SNO
    37310  14344        DSZ     ONF         ;MORE LINE NUMBERS ?
    37311    410        JMP     LSTG2       ;   YES
    37312  30355        LDA     2,.BUS      ;   NO
    37313  24333        LDA     1,TSE
    37314  36004        LDA     3,@PIB
    37315 166400        SUB     3,1
    37316 125540        INCOL   1,1
    37317  45003        STA     1,PBC.,2
    37320    407        JMP     LISTA

    37321  30236 LSTG2: LDA     2,C254      ;OUTPUT COMMA AND SPACE
    37322   6132        OUTBYTE
    37323  30056        LDA     2,C240
    37324   6132        OUTBYTE
    37325  10333        ISZ     TSE         ;STEP LINE NUMBER POINTER
    37326    760        JMP     LSTG1


    37327  34355 LISTA: LDA     3,.BUS      ;END OF STATEMENT
    37330  31402        LDA     2,PLC.,3
    37331  25401        LDA     1,VDT.,3
    37332  21406        LDA     0,UVS.,3
    37333 146033        SLS     2,1         ;LAST LINE OF PROGRAM ?
    37334    403        JMP     LSTA1       ;   YES
    37335 173000        ADD     3,2         ;   NO
    37336  21001        LDA     0,1,2       ;DISPLACEMENT TO NEXT LINE
    37337  24065 LSTA1: LDA     1,C400
    37340 123120        ADDZL   1,0         ;BYTE DISPLACEMENT OF NEXT LINE
    37341  25403        LDA     1,PBC.,3
    37342 122033        SLS     1,0         ;POSSIBLE COMMENT ?
    37343    541        JMP     LISTE       ;   NO
    37344   6144        XGETBYTE            ;   YES
    37345 151015        SNZ     2,2         ;ANY COMMENT ?
    37346    536 JLSTE: JMP     LISTE       ;   NO
    37347   6133        OUTTEXT             ;   YES
    37350 120241        .TXTF   ; !
    37351 120000  ;

    37352    657        JMP     LISTR       ;OUTPUT THE COMMENT
```

```
;                       << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE *******
 37353  10341 LISTZ: ISZ     TSE6        ;VARIABLE NUMBER - WHAT KIND ?
 37354    417         JMP     LISTV       ;   NAMED VARIABLE
 37355  50341         STA     2,TSE6      ;   "DEF" DUMMY VARIABLE
 37356  34043         LDA     3,C37
 37357 173700         ANDS    3,2         ;DUMMY VARIABLE NAME WORD
 37360  34355         LDA     3,.BUS
 37361  21401         LDA     0,VDT.,3
 37362 117000         ADD     0,3         ;.VDT
 37363  51400         STA     2,0,3
 37364  30234         LDA     2,C250      ;OUTPUT "("
 37365   6132         OUTBYTE
 37366  30341         LDA     2,TSE6      ;OUTPUT DUMMY NAME
 37367   6132         OUTBYTE
 37370  30235         LDA     2,C251      ;OUTPUT ")"
 37371    670         JMP     LISTS


 37372  17400         17400
 37373  34246 LISTV: LDA     3,C335      ;VARIABLE, NOT "DEF" DUMMY
 37374 156422         SUBZ    2,3,SZC     ;NAMED DUMMY VARIABLE ?
 37375    404         JMP     .+4         ;   NO
 37376  30057         LDA     2,C244      ;   YES
 37377 172400         SUB     3,2         ;CHANGE NUMBER TO "&" OR "%"
 37400    661 JLSTS: JMP     LISTS

 37401  24052         LDA     1,C177      ;ORDINARY PROGRAM VARIABLE
 37402 147520         ANDZL   2,1
 37403  34355         LDA     3,.BUS
 37404  31401         LDA     2,VDT.,3
 37405 173000         ADD     3,2         ;.VDT
 37406 133000         ADD     1,2
 37407  21000         LDA     0,0,2       ;VARIABLE NAME WORD
 37410  40335         STA     0,TSE2
 37411  24761         LDA     1,LISTV-1
 37412 107700         ANDS    0,1         ;FIRST LETTER OF NAME
 37413  30062         LDA     2,C300
 37414 133000         ADD     1,2         ;CHANGE TO ASCII
 37415   6132         OUTBYTE             ;OUTPUT LETTER
 37416  30335         LDA     2,TSE2
 37417  24050         LDA     1,C77
 37420 133405         AND     1,2,SNR     ;DIGIT INCLUDED IN NAME ?
 37421    404         JMP     .+4         ;   NO
 37422  24053         LDA     1,C200      ;   YES
 37423 133000         ADD     1,2         ;CHANGE TO ASCII
 37424   6132         OUTBYTE             ;OUTPUT DIGIT
 37425  24335         LDA     1,TSE2
 37426  30057         LDA     2,C244
 37427 127112         ADDL#   1,1,SZC     ;STRING VARIABLE ?
 37430   6132         OUTBYTE             ;   YES, OUTPUT "$"
 37431    631 JLSS1: JMP     LISTS+1
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******** THIS PAGE IS GARBAGE  ******
    37432    6331 LISTD: NEXTBYTE          ;"DATA" STATEMENT
    37433   20050        LDA     0,C77
    37434  143400        AND     2,0
    37435   40336        STA     0,TSE3     ;DATA PRECISION
    37436   24002        LDA     1,C2
    37437  106415        SNE     0,1        ;PRECISION 2 ?
    37440     407        JMP     LSTD1-6    ;   YES
    37441   30060        LDA     2,C260     ;   NO
    37442  113000        ADD     0,2
    37443    6132        OUTBYTE
    37444    6133        OUTTEXT
    37445  122654        .TXTF   ;%,
    37446       0 ;

    37447   34355        LDA     3,.BUS     ;SET ABSOLUTE WORD POINTER
    37450   31403        LDA     2,PBC.,3
    37451  151220        MOVZR   2,2
    37452   22004        LDA     0,@PIB
    37453  113000        ADD     0,2
    37454   50335        STA     2,TSE2
    37455  102520 LSTD1: SUBZL   0,0        ;GET A DATA ELEMENT
    37456   24336        LDA     1,TSE3
    37457   30335        LDA     2,TSE2
    37460  155000        MOV     2,3
    37461  137000        ADD     1,3
    37462   54335        STA     3,TSE2
    37463    6120        DECIMAL            ;LOAD DATA ELEMENT
    37464   20030        LDA     0,C10
    37465  126400        SUB     1,1        ;OUTPUT IT
    37466    6120        DECIMAL
    37467   14334        DSZ     TSE1       ;END OF STATEMENT ?
    37470     402        SKIP
    37471     413        JMP     LISTE      ;   YES
    37472   30005        LDA     2,RUP      ;   NO
    37473   25005        LDA     1,OBP.,2
    37474   20236        LDA     0,C254     ;COMMA REPLACES LAST SPACE
    37475    6134        PUTBYTE
    37476     757        JMP     LSTD1

    37477    6331 LISTO: NEXTBYTE          ;"ON" STATEMENT
    37500   50344        STA     2,ONF
    37501     730        JMP     JLSS1

    37502   36531        LWRTE
    37503   36700        LISTT
    37504  126400 LISTE: SUB     1,1        ;END OF STATEMENT
    37505    6775        JSR     @.-3       ;OUTPUT THE LINE
    37506   20073        LDA     0,ESCF
    37507  101014        SKZ     0,0        ;USER PRESS ESC ?
    37510    2773        JMP     @LISTE-1   ;   YES, STOP LISTING
    37511    2301        JMP     @.DCOD     ;   NO, LIST NEXT LINE
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE ******
 37512       0          0
 37513   30355 FSLN:  LDA     2,.BUS     ;FIND STARTING LINE NUMBER
 37514   25001         LDA     1,VDT.,2
 37515  147000         ADD     2,1
 37516   44774         STA     1,FSLN-1   ;.VDT
 37517   25005         LDA     1,SLT.,2
 37520  133000         ADD     1,2
 37521   25000         LDA     1,0,2
 37522  106432         SGR     0,1        ;LINE NUMBER >= (A0) ?
 37523    1401         JMP     1,3        ;   YES
 37524  151400         INC     2,2        ;   NO
 37525  151400         INC     2,2
 37526   24764         LDA     1,FSLN-1
 37527  146032         SGE     2,1        ;END OF SLT ?
 37530     771         JMP     .-7        ;   NO
 37531    1400         JMP     0,3        ;NO STARTING LINE NUMBER


 37532   50340 LISTU: STA     2,TSE5     ;USER DEFINED FUNCTION
 37533   30032         LDA     2,C12
 37534    6326         JSR     @.WORD     ;OUTPUT "FN "
 37535   34005         LDA     3,RUP
 37536   15405         DSZ     OBP.,3     ;DELETE LAST SPACE
 37537   30340         LDA     2,TSE5
 37540   24053         LDA     1,C200
 37541  133000         ADD     1,2        ;FUNCTION LETTER IN ASCII
 37542     636         JMP     JLSTS


 37543   24244 LISTW: LDA     1,C331     ;OPERATOR WORD
 37544     431         JMP     LISTF+1


 37545   24003 LISTK: LDA     1,C3       ;FLOATING POINT CONSTANT
 37546  147400         AND     2,1
 37547  121400         INC     1,0
 37550  125014         SKZ     1,1
 37551  125400         INC     1,1
 37552   34355         LDA     3,.BUS
 37553   31403         LDA     2,PBC.,3
 37554  151620         INCZR   2,2
 37555  143120         ADDZL   2,0
 37556   41403         STA     0,PBC.,3
 37557  102520         SUBZL   0,0
 37560   36004         LDA     3,@PIB
 37561  173000         ADD     3,2
 37562    6120         DECIMAL
 37563   20030         LDA     0,C10
 37564  126400         SUB     1,1
 37565    6120         DECIMAL
 37566   34005         LDA     3,RUP
 37567   15405         DSZ     OBP.,3
 37570     641         JMP     JLSS1
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;****** THIS PAGE IS GARBAGE *******
    37571 145000 LISTI: MOV     2,1         ;INTEGER CONSTANT
    37572   6324        JSR     @.SNO
    37573    636        JMP     JLSS1


    37574  24213 LISTF: LDA     1,C60       ;PRIMITIVE FUNCTION
    37575 132400        SUB     1,2
    37576   6326        JSR     @.WORD
    37577    632        JMP     JLSS1


    37600      2        .BLK 2
    37602  50777 LISTC: STA     2,.-1       ;LIST SPECIAL CHARACTER
    37603  30063        LDA     2,C334
    37604   6132        OUTBYTE             ;OUTPUT "\"
    37605  30774        LDA     2,LISTC-1
    37606 151300        MOVS    2,2
    37607 145220        MOVZR   2,1
    37610  30060 LSTC2: LDA     2,C260
    37611  34003        LDA     3,C3
    37612  54767        STA     3,LISTC-1
    37613 125040        MOVO    1,1
    37614 102460        SUBC    0,0
    37615 125120        MOVZL   1,1         ;SHIFT OUT A DIGIT
    37616 101100        MOVL    0,0
    37617 125120        MOVZL   1,1
    37620 101100        MOVL    0,0
    37621 125120        MOVZL   1,1
    37622 101100        MOVL    0,0
    37623 113000        ADD     0,2         ;CONVERT TO ASCII
    37624  44754        STA     1,LISTC-2
    37625   6132        OUTBYTE
    37626  24752        LDA     1,LISTC-2
    37627  30060        LDA     2,C260
    37630 125020        MOVZ    1,1
    37631  14750        DSZ     LISTC-1     ;DONE ?
    37632    761        JMP     LSTC2+3     ;   NO
    37633  30063 LSTC1: LDA     2,C334      ;   YES
    37634   6132        OUTBYTE             ;OUTPUT "\"
    37635   2401        JMP     @.+1
    37636  37232        LISTR+1
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>

;****** THIS PAGE IS GARBAGE ******
;   "BUSINESS PROTECT" SOURCE #6 OF 6 TAPES FOR "IRIS" R8.2

;THIS CODE ENCODES THE TERMINAL CONTROL MNEMONICS FOUND BETWEEN PAIRS OF
;SINGLE QUOTES


37637      0          0
37640  30270 ESQ:    LDA     2,C375    ;ENCODE SINGLE QUOTE
37641   6332          SAVEBYTE          ;STORE LEADING QUOTE
37642   6125 EMN:     INBYTE            ;GET FIRST BYTE OF MNEMONIC
37643  20233          LDA     0,C247
37644 112415          SNE     0,2       ;END OF TERMINAL CONTROL FUNCTION ?
37645    415          JMP     EMDON     ;   YES
37646  50771          STA     2,ESQ-1   ;   NO, SAVE BYTE
37647   6125          INBYTE            ;GET NEXT BYTE
37650  24767          LDA     1,ESQ-1   ;RETRIEVE FIRST BYTE
37651 125300          MOVS    1,1       ;ASSEMBLE TWO BYTE MNEMONIC
37652 147000          ADD     2,1       ;INTO A1
37653  30522          LDA     2,.MN
37654   4441          JSR     FIND      ;LOOKUP MNEMONIC IN TABLE
37655  34454          LDA     3,.CD     ;GET ADDRESS OF CODE
37656 117000          ADD     0,3       ;CORRESPONDING TO THE MNEMONIC
37657  31400          LDA     2,0,3     ;FETCH THE CODE
37660   6332          SAVEBYTE          ;STORE EDITED BYTE
37661    761          JMP     EMN       ;DO NEXT MNEMONIC

37662  30270 EMDON:  LDA     2,C375
37663   6332          SAVEBYTE          ;STORE TRAILING QUOTE
37664   2303          JMP     @.EDEX    ;CONTINUE
```

```
;                        << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;******* THIS PAGE IS GARBAGE ********
;THIS CODE DECODES THE TERMINAL CONTROL CODES BACK INTO THE APPROPRIATE
;MNEMONICS


37665       0          0
37666   30233 DSQ:    LDA     2,C247    ;DECODE SINGLE QUOTE
37667    6132          OUTBYTE           ;OUTPUT LEADING QUOTE
37670    6331 DCD:     NEXTBYTE          ;GET FIRST CODE
37671   20270          LDA     0,C375
37672  112415          SNE     0,2       ;LAST CODE ALREADY DONE ?
37673     416          JMP     DCDON     ;   YES
37674  145000          MOV     2,1
37675   30434          LDA     2,.CD
37676    4417          JSR     FIND      ;LOOKUP CODE IN TABLE
37677    6142          TRAPFAULT         ;   NOT FOUND !?
37700   30475          LDA     2,.MN     ;GET ADDRESS OF MNEMONIC
37701  113000          ADD     0,2       ;CORRESPONDING TO THE CODE
37702   31000          LDA     2,0,2
37703   50762          STA     2,DSQ-1   ;SAVE IT
37704  151300          MOVS    2,2
37705    6132          OUTBYTE           ;OUTPUT LEFT BYTE OF MNEMONIC
37706   30757          LDA     2,DSQ-1   ;RETRIEVE MNEMONIC
37707    6132          OUTBYTE           ;OUTPUT RIGHT BYTE OF MNEMONIC
37710     760          JMP     DCD       ;DO NEXT CODE

37711   30233 DCDON:  LDA     2,C247
37712    6132          OUTBYTE           ;OUTPUT TRAILING QUOTE
37713    2314          JMP     @.LSTX    ;CONTINUE
```

```
;                        << SI = R94PROTECTSC;  BO = 1/A.SPP4.9062! >>
;FIND == PROCEDURE TO DO A LINEAR TABLE SEARCH

;ON ENTRY:   A1 = VALUE TO FIND IN TABLE
;            A2 = ADDRESS OF FIRST ENTRY OF TABLE

;THE TABLE MUST BE TERMINATED WITH A ZERO

;CALLING SEQUENCE:

;       JSR     FIND
;            ...             ;RETURN IF NOT FOUND
;            ...             ;RETURN IF FOUND

;ON RETURN:   A0 = DISPLACEMENT INTO TABLE AT WHICH THE VALUE WAS FOUND


37714       0          0
37715   54777 FIND:   STA     3,FIND-1 ;SAVE RETURN ADR
37716  102400          SUB     0,0      ;INITIALIZE DISPLACEMENT COUNTER
37717   35000 LOOK:   LDA     3,0,2    ;FETCH A TABLE ENTRY
37720  175015          SNZ     3,3      ;IS IT THE END OF THE TABLE ?
37721    2773          JMP     @FIND-1  ;   YES, DO A NON-SKIP RETURN
37722  166415          SNE     3,1      ;IS IT WHAT WE'RE LOOKING FOR ?
37723     404          JMP     FOUND    ;   YES
37724  101400          INC     0,0      ;   NO, BUMP DISPLACEMENT COUNTER
37725  151400          INC     2,2      ;       AND ADDRESS OF CURRENT ENTRY
37726     771          JMP     LOOK     ;TRY AGAIN

37727   34765 FOUND:  LDA     3,FIND-1 ;VALUE FOUND
37730    1401          JMP     1,3      ;DO A SKIP RETURN
```

```
;                       << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;THIS IS THE TABLE OF OCTAL CODES CORRESPONDING TO THE TERMINAL
;CONTROL MNEMONICS IN THE TABLE ON THE NEXT PAGE

     37731   37732 .CD:   .+1
     37732       3        3        ;'ET'
     37733       7        7        ;'RB'
     37734      10       10        ;'ML'
     37735      12       12        ;'LF'
     37736      13       13        ;'VT'
     37737      14       14        ;'FF'
     37740      15       15        ;'CR'
     37741      17       17        ;'MH'
     37742      20       20        ;'CS'
     37743      40       40        ;'MR'
     37744      41       41        ;'RD'
     37745      43       43        ;'CU'
     37746      44       44        ;'CL'
     37747      45       45        ;'CE'
     37750      52       52        ;'MD'
     37751      53       53        ;'MU'
     37752      60       60        ;'BB'
     37753      61       61        ;'EB'
     37754      62       62        ;'BR'
     37755      63       63        ;'ER'
     37756      64       64        ;'BD'
     37757      65       65        ;'ED'
     37760      66       66        ;'BP'
     37761      67       67        ;'EP'
     37762      70       70        ;'BU'
     37763      71       71        ;'EU'
     37764      72       72        ;'BX'
     37765      73       73        ;'EX'
     37766      74       74        ;'FM'
     37767      75       75        ;'FX'
     37770      76       76        ;'LK'
     37771      77       77        ;'UK'
     37772     100      100        ;'BT'
     37773     101      101        ;'MP'

     37774       0        0        ;TABLE TERMINATOR
```

```
;                       << SI = R94PROTECTSC; BO = 1/A.SPP4.9062! >>
;THIS IS THE TABLE OF TERMINAL CONTROL MNEMONICS

              200 P=     200     ;PARITY BIT

    37775   37776 .MN:   .+1
    37776  142724        "E+P*K+"T+P
    37777  151302        "R+P*K+"B+P
    40000  146714        "M+P*K+"L+P
    40001  146306        "L+P*K+"F+P
    40002  153324        "V+P*K+"T+P
    40003  143306        "F+P*K+"F+P
    40004  141722        "C+P*K+"R+P
    40005  146710        "M+P*K+"H+P
    40006  141723        "C+P*K+"S+P
    40007  146722        "M+P*K+"R+P
    40010  151304        "R+P*K+"D+P
    40011  141725        "C+P*K+"U+P
    40012  141714        "C+P*K+"L+P
    40013  141705        "C+P*K+"E+P
    40014  146704        "M+P*K+"D+P
    40015  146725        "M+P*K+"U+P
    40016  141302        "B+P*K+"B+P
    40017  142702        "E+P*K+"B+P
    40020  141322        "B+P*K+"R+P
    40021  142722        "E+P*K+"R+P
    40022  141304        "B+P*K+"D+P
    40023  142704        "E+P*K+"D+P
    40024  141320        "B+P*K+"P+P
    40025  142720        "E+P*K+"P+P
    40026  141325        "B+P*K+"U+P
    40027  142725        "E+P*K+"U+P
    40030  141330        "B+P*K+"X+P
    40031  142730        "E+P*K+"X+P
    40032  143315        "F+P*K+"M+P
    40033  143330        "F+P*K+"X+P
    40034  146313        "L+P*K+"K+P
    40035  152713        "U+P*K+"K+P
    40036  141324        "B+P*K+"T+P
    40037  146720        "M+P*K+"P+P

    40040       0        0       ;TABLE TERMINATOR

                0        .ERR    BPS+7000<.;Overlap check

            41200 EBS=   BPS+7000          ;EDITOR BUFFER STORAGE

                         .EOT    ; "PROTECT" R9.4 SOURCE #6
```

```
;                    << SI = R94SPP4S6A; BO = 1/A.SPP4.9062! >>

                     ;
                     ;     Key translation table
                     ;

          42200              .LOC    EBS+1000
          42200    PLTBL:
    42200 15264              015264
    42201 16755              016755
    42202 16126              016126
    42203 17163              017163
    42204 16352              016352
    42205 16243              016243
    42206 16600              016600
    42207 15363              015363
    42210 15726              015726
    42211 17767              017767
    42212 14756              014756
    42213 17751              017751
    42214 15744              015744
    42215 20003              020003
    42216 16273              016273
    42217 17013              017013
    42220 16374              016374
    42221 16453              016453
    42222 15444              015444
    42223 16011              016011
    42224 15160              015160
    42225 17165              017165
    42226 15023              015023
    42227 17421              017421
    42230 16200              016200
    42231 17303              017303
    42232 16657              016657
    42233 15167              015167
    42234 16526              016526
    42235 15533              015533
    42236 16106              016106
    42237 17417              017417
    42240 15117              015117
    42241 17155              017155
    42242 16322              016322
    42243 16373              016373
    42244 15355              015355
    42245 16215              016215
    42246 16602              016602
    42247 15403              015403
    42250 15123              015123
    42251 17171              017171
    42252 14762              014762
    42253 20175              020175
```

; << SI = R94SPP4S6A; BO = 1/A.SPP4.9062! >>

42254	15033	015033
42255	17531	017531
42256	15226	015226
42257	16575	016575
42260	16100	016100
42261	17563	017563
42262	15612	015612
42263	14711	014711
42264	14723	014723
42265	20031	020031
42266	15547	015547
42267	15435	015435
42270	15760	015760
42271	20047	020047
42272	15644	015644
42273	15001	015001
42274	15153	015153
42275	17101	017101
42276	15163	015163
42277	17345	017345
42300	16670	016670
42301	15063	015063
42302	16237	016237
42303	16677	016677
42304	15626	015626
42305	15101	015101
42306	14775	014775
42307	20071	020071
42310	16474	016474
42311	16137	016137
42312	16642	016642
42313	15113	015113
42314	15555	015555
42315	15311	015311
42316	15302	015302
42317	16531	016531
42320	15666	015666
42321	15025	015025
42322	16055	016055
42323	17713	017713
42324	16413	016413
42325	15763	015763
42326	15017	015017
42327	17655	017655
42330	15423	015423
42331	15725	015725
42332	15653	015653
42333	15225	015225
42334	15277	015277
42335	16615	016615

; << SI = R94SPP4S6A; BD = 1/A.SPP4.9062! >>

42336	16606	016606
42337	15477	015477
42340	16050	016050
42341	17673	017673
42342	16467	016467
42343	16063	016063
42344	15646	015646
42345	15221	015221
42346	15654	015654
42347	15045	015045
42350	15215	015215
42351	16515	016515
42352	16267	016267
42353	16753	016753
42354	16000	016000
42355	20147	020147
42356	14736	014736
42357	20205	020205
42360	16213	016213
42361	16763	016763
42362	14774	014774
42363	20265	020265
42364	15522	015522
42365	15255	015255
42366	16664	016664
42367	15177	015177
42370	15367	015367
42371	16171	016171
42372	15607	015607
42373	15341	015341
42374	15745	015745
42375	20127	020127
42376	15504	015504
42377	15611	015611
42400	16655	016655
42401	15007	015007
42402	14746	014746
42403	20055	020055
42404	16372	016372
42405	16153	016153
42406	15630	015630
42407	14731	014731
42410	16035	016035
42411	17743	017743
42412	15217	015217
42413	17031	017031
42414	16515	016515
42415	15563	015563
42416	15234	015234
42417	16621	016621

```
;                       << SI = R94SPP4S6A;  BO = 1/A. SPP4. 9062! >>

    42420   15070         015070
    42421   17745         017745
    42422   15563         015563
    42423   15345         015345
    42424   16365         016365
    42425   16263         016263
    42426   15175         015175
    42427   17341         017341
    42430   15530         015530
    42431   15525         015525
    42432   15410         015410
    42433   16315         016315
    42434   15343         015343
    42435   16445         016445
    42436   16076         016076
    42437   17623         017623
    42440   15734         015734
    42441   20167         020167
    42442   15055         015055
    42443   17711         017711
    42444   15365         015365
    42445   16261         016261
    42446   16535         016535
    42447   15467         015467
    42450   15450         015450
    42451   15755         015755
    42452   16124         016124
    42453   17277         017277
    42454   16364         016364
    42455   16157         016157
    42456   15660         015660
    42457   14751         014751
    42460   16656         016656
    42461   15157         015157
    42462   14726         014726
    42463   17765         017765
    42464   15356         015356
    42465   16455         016455
    42466   15525         015525
    42467   15511         015511
    42470   15552         015552
    42471   15451         015451
    42472   15622         015622
    42473   15021         015021
    42474   15533         015533
    42475   15565         015565
    42476   15360         015360
    42477   16311         016311
    42500 177777          -1      ;End of table

                          .EOT    ;   'SPP4' R9.2 SOURCE #8
```

```
;                       << SI = R94SPP4S7B; BO = 1/A.SPP4.9062! >>

;  'PROTECT' SOURCE #7 FOR POINT 4 IN-HOUSE USE ONLY

            42501 SPP4  =
            35777       .LOC    POKEY
     35777  43177       PNC
            42501       .LOC    SPP4

;-------------------------------------------------------------
;
;  THIS ROUTINE ADDS THE LEVEL 2 PROTECT KEY & ADDITIONAL
; LIST PROTECTION TO A POINT 4 BASIC APPLICATION PROGRAM.
; THE FORMAT OF THE PROTECT STATEMENT ENTERED BY THE
; OPERATOR IS AS FOLLOWS:
;
;    PROTECT  [PROGRAM NAME],[KEY],[LIST PROTECT OPTIONS]
;
; BRACKETS INDICATE AN OPTIONAL PARAMETER.
; THE PROGRAM BEING PROTECTED MUST BE IN THE USER AREA.
; THE KEY ENTERED IS IN DECIMAL & IS THE ACTUAL LEVEL 2 KEY.
; THE LIST PROTECT OPTIONS WHICH MAY BE ENTERED ARE:
;    1 = CONVERT GOTO'S TO GOSUB'S, & VICE VERSA.
;    2 = REVERSE THE RELATIONAL OPERATOR OF ALL 'IF' STATEMENTS.
;    3 = DO BOTH 1 & 2 ABOVE.
;    DEFAULT = NONE OF THE ABOVE.
; THE LIST PROTECT BITS ARE SET IN THE FLAG WORD & THE FLAG
; WORD IS XOR'ED WITH THE CORRESPONDING BITS IN THE STATUS:
;   FLAG WORD BIT 7 = REVERSE 'IF' OPERATOR
;     "    "     "  12= SWAP GOTO'S & GOSUB'S
;
;-------------------------------------------------------------


    42501       0 TCH:  0       ;TERMINATOR
    42502   36235 .BADK:BADK
    42503   36312 .RIBP:RIBP
    42504   36265 .SKEY:SKEY
    42505   36340 .CALK:CALCK
    42506   36252 .OKEY:OKEY
    42507     262 C262: 262
    42510   14711 VKOFF:14711   ;Offset used to disguise key xlat table
    42511   36203 .CKEY:CKEY    ;Address of normal PROTECT entry
    42512   42200 .PLTBL:       PLTBL     ;Physical to Logical key xlat table
    42513   43001 .PLNR:PLNR

    42514  102414 EP6:  EBS+6*2
    42515       0 LPO:  0       ;LIST PROTECT OPTIONS:
;                           ;  -1 = SWAP GOTO'S & GOSUB'S
;                           ;   0 = REVERSE 'IF' RELATIONAL OPERATORS
;                           ;   1 = BOTH OF THE ABOVE
```

```
;                    << SI = R94SPP4S7B;  BO = 1/A.SPP4.9062! >>

 42516   6766 P4K:   JSR     @.SKEY    ;SCAN INPUT BUFFER FOR KEY
 42517  20026       LDA     0,C6
 42520 126400       SUB     1,1
 42521  46765       STA     1,@.OKEY
 42522   6120       DECIMAL;CONVERT IOB TO F.P. DECIMAL
 42523    406       JMP     NOKEY     ;NO KEY
 42524  50755       STA     2,TCH     ;SAVE TERMINATOR
 42525   6121       FIX     ;CONVERT DA TO BINARY
 42526   2754       JMP     @.BADK    ;> 65535
 42527  46757       STA     1,@.OKEY
 42530  30751       LDA     2,TCH
 42531  20236 NOKEY:LDA     0,C254    ;COMMA
 42532 142414       SEQ     2,0       ;LIST PROTECT OPTION ?
 42533    465       JMP     ADK       ;   NO
 42534 102000       ADC     0,0       ;   YES
 42535   6126       INSTBYTE          ;GET OPTION
 42536  20751       LDA     0,C262
 42537 112400       SUB     0,2
 42540 150015       COM#    2,2,SNR   ;Validate L.P. option
 42541    427       JMP     VLP       ;Good
 42542 151235       MOVZR#  2,2,SNR
 42543    425       JMP     VLP       ;Good
 42544   6133 NVLP: OUTTEXT
 42545 106711       .TXTF   "<215>I
 42546 167366 nv
 42547 160754 al
 42550 164744 id
 42551 120314  L
 42552 164763 is
 42553 172240 t
 42554 150362 Pr
 42555 167764 ot
 42556 162743 ec
 42557 172240 t
 42560 147760 Op
 42561 172351 ti
 42562 167756 on
 42563 171607 s<207>
 42564      0 "

 42565   6141       STOUTPUT
 42566   6101       CALL
 42567 100000       SCOPE
 42570 100010 VLP:  NOP
 42571  50724       STA     2,LPO
 42572  30355       LDA     2,.BUS    ;PREPARE FOR CONVERSION
 42573 126520       SUBZL   1,1
 42574  45075       STA     1,TSU.+2,2
 42575  25005       LDA     1,SLT.,2
 42576  45002       STA     1,PLC.,2
 42577   6714       JSR     @.PLNR    ;CONVERT
 42600    402       JMP     SLPB      ;DONE, SET LIST PROTECT BITS
 42601    776       JMP     .-2       ;CONTINUE CONVERSION
```

```
;                        << SI = R94SPP4S7B;  BO = 1/A.SPP4.9062! >>
    42602   20713 SLPB:  LDA     0,LPO      ;SET LIST PROTECT BITS IN FLAG
    42603    4404        JSR     .+4
    42604   10000        10000   ;SWAP GOTO'S & GOSUBS
    42605     200        200     ;REVERSE IF OPERATORS
    42606   10200 P4M:   10200   ;BOTH
    42607  117000        ADD     0,3
    42610   25401        LDA     1,1,3
    42611   30355        LDA     2,.BUS
    42612   21024        LDA     0,FLAG.,2
    42613  135000        MOV     1,3        ;XOR FLAG WITH LP BITS
    42614  117520        ANDZL   0,3
    42615  107000        ADD     0,1
    42616  166400        SUB     3,1
    42617   45024        STA     1,FLAG.,2;SAVE FLAG WORD
    42620   22666 ADK:   LDA     0,@.OKEY ;ADD LEVEL 2 PROTECT KEY
    42621  101015        SNZ     0,0        ;KEY ?
    42622    2661        JMP     @.RIBP     ;   NO
    42623    4474        JSR     ENCD2      ;   YES, GENERATE FAKE 'REM' + CHECKSUM
    42624   20437        LDA     0,CLNO     ;PREPARE TO CALL 'EDML'...
    42625   40346        STA     0,LNO      ;LINE #
    42626   20666        LDA     0,EP6
    42627   40353        STA     0,EBP
    42630   34661        LDA     3,.CKEY
    42631   54306        STA     3,.EDUN    ;RETURN ADDRESS
    42632  141000        MOV     2,0        ;A(1ST SOURCE WORD)
    42633   24025        LDA     1,C5
    42634  107000        ADD     0,1        ;A(LAST SOURCE WORD)
    42635   30206        LDA     2,.EBS     ;A(DESTINATION)
    42636    6101        CALL               ;MOVE FAKE 'REM' TO EBS
    42637  100015        MOVEWORDS

    42640   34652        LDA     3,.PLTBL ;Load addr of physical to logical key xlat table
    42641   24647        LDA     1,VKOFF  ;Load key offset value
    42642   32644        LDA     2,@.OKEY ;Load key value
    42643  133000        ADD     1,2      ;Add offset to match value in table
    42644   21400 XPLNK: LDA     0,0,3    ;Load next physical key
    42645  100015        COM#    0,0,SNR  ;End of table?
    42646     676        JMP     NVLP     ;   No, No such key, Print error message
    42647  112415        SNE     0,2      ;Found physical key?
    42650     404        JMP     XPLXK    ;   Yes, Translate key to logical key
    42651  175400        INC     3,3      ;   No, Try next key
    42652  175400        INC     3,3
    42653     771        JMP     XPLNK

    42654   21401 XPLXK: LDA     0,1,3    ;Load logical key value
    42655  122400        SUB     1,0      ;Subtract offset
    42656  101220        MOVZR   0,0      ;Shift into proper position
    42657   42627        STA     0,@.OKEY ;Set key to logical key value
    42660   34002        LDA     3,C2     ;INSERT LINE INDICATION
    42661  102000        ADC     0,0      ;CAN'T = 0
    42662    2304        JMP     @.EDML   ;MOVE FAKE 'REM' TO USER AREA
```

```
;                        << SI = R94SPP4S7B; BO = 1/A.SPP4.9062! >>

;-----------------------------------------------------------------
;
;GENERATE 6 WORDS & A CHECKSUM TO BE ENCODED INTO THE BASIC
;PROGRAM AS THE "POINT 4" PROTECT MECHANISM.
;
;INPUT:
;    A0 = LEVEL 2 KEY
;    A3 = RETURN
;
;OUTPUT:
;    A2 = ADDRESS OF THE FOLLOWING 6 WORD TABLE:
;
;          WORD 0:  53240
;           "    1:  XOR MASK
;           "    2:  LEVEL 2 KEY (XOR'ED WITH MASK)
;           "    3:  RANDOM #
;           "    4:  CHECKSUM
;           "    5:  0
;
;    NOTE:
;     1.  THE XOR MASK IS USED ONLY ON THE LEVEL 2 KEY.
;     2.  WORDS 1-4 CANNOT CONTAIN A NULL BYTE.
;     3.  THE SUM OF WORDS 0-4 MUST = 123420 (OCTAL).
;
;-----------------------------------------------------------------


; Generate noise in Ac0.  This will be affected by the SLT & the previous
; noise word.

    42663 123420 CLNO:  123420 ;Fake line #
    42664      0        0
    42665  54777 RNDM1: STA    3,.-1
    42666  30355        LDA    2,.BUS
    42667  25005        LDA    1,SLT.,2
    42670 133000        ADD    1,2        ;A(SLT)
    42671  34355        LDA    3,.BUS
    42672  25401        LDA    1,VDT.,3
    42673 137020        ADDZ   1,3        ;A(VDT)
    42674  20410        LDA    0,PRN      ;Previous noise word
    42675  25000 ASLT:  LDA    1,0,2      ;Accumulate SLT
    42676 123200        ADDR   1,0
    42677 151400        INC    2,2
    42700 156032        SGE    2,3        ;End of SLT ?
    42701    774        JMP    ASLT       ;  No
    42702  40402        STA    0,PRN      ;Save for next time
    42703   2761        JMP    @RNDM1-1   ;Return

    42704      0 PRN:   0                 ;Previous noise word
```

```
;                        << SI = R94SPP4S7B;  BO = 1/A.SPP4.9062!  >>
42705   42665 .RNDM: RNDM1
42706   42707 .ECR2: ECR2
42707   53240 ECR2:  53240   ;RESULTS
42710       4        .BLK    4
42714       0        0       ;TERMINATOR
42715       0 L2KEY: 0       ;LEVEL 2 KEY
42716       0 RTN2:  0       ;RETURN
42717   54777 ENCD2: STA     3,RTN2
42720   40775        STA     0,L2KEY
42721    6764 ECL1:  JSR     @.RNDM    ;GET RANDOM # IN A0
42722   30764        LDA     2,.ECR2
42723   41001        STA     0,1,2     ;WORD 1:  XOR MASK
42724   24771        LDA     1,L2KEY
42725  135000        MOV     1,3       ;XOR MASK & KEY
42726  117520        ANDZL   0,3
42727  107000        ADD     0,1
42730  166400        SUB     3,1
42731   45002        STA     1,2,2     ;WORD 2:  KEY (XOR'ED)
42732    6753 ECL2:  JSR     @.RNDM
42733   30753        LDA     2,.ECR2
42734   41003        STA     0,3,2     ;WORD 3:  RANDOM #
42735   25000        LDA     1,0,2
42736  123000        ADD     1,0       ;CALC CHECKSUM
42737   25001        LDA     1,1,2
42740  123000        ADD     1,0
42741   25002        LDA     1,2,2
42742  123000        ADD     1,0
42743   24720        LDA     1,CLNO
42744  106400        SUB     0,1
42745   45004        STA     1,4,2     ;WORD 4:  CHECKSUM
42746   24064        LDA     1,C377    ;CHECK FOR NULL BYTES
42747   21001        LDA     0,1,2     ;  IN WORD 1
42750  107415        AND#    0,1,SNR
42751     750        JMP     ECL1      ;NULL FOUND, START OVER
42752  101300        MOVS    0,0
42753  107415        AND#    0,1,SNR
42754     745        JMP     ECL1      ;NULL FOUND
42755   21002        LDA     0,2,2     ;  IN WORD 2...
42756  107415        AND#    0,1,SNR
42757     742        JMP     ECL1
42760  101300        MOVS    0,0
42761  107415        AND#    0,1,SNR
42762     737        JMP     ECL1
42763   21003        LDA     0,3,2     ;  IN WORD 3...
42764  107415        AND#    0,1,SNR
42765     745        JMP     ECL2
42766  101300        MOVS    0,0
42767  107415        AND#    0,1,SNR
42770     742        JMP     ECL2
42771   21004        LDA     0,4,2     ;  IN WORD 4...
42772  107415        AND#    0,1,SNR
42773     737        JMP     ECL2
42774  101300        MOVS    0,0
42775  107415        AND#    0,1,SNR
42776     734        JMP     ECL2
42777    2717        JMP     @RTN2     ;NO NULLS, RETURN
```

```
;                    << SI = R94SPP4S7B; BO = 1/A.SPP4.9062! >>

43000     140 C140:  140

43001   54333 PLNR:  STA     3,TSE      ;FIND GOTO, GOSUB, & IF
43002   34355        LDA     3,.BUS
43003   54040        STA     3,SBA      ;FORCE REFERENCES TO PROG AREA
43004   30355 PLNR1: LDA     2,.BUS
43005   35002        LDA     3,PLC.,2
43006   25001        LDA     1,VDT.,2
43007  166033        SLS     3,1        ;END OF PROGRAM ?
43010    2333        JMP     @TSE       ;  YES, NON-SKIP RETURN
43011  157000        ADD     2,3        ;  NO
43012   21401        LDA     0,1,3
43013  101120        MOVZL   0,0        ;SET BYTE POINTER
43014   41003        STA     0,PBC.,2   ;POINT TO FRONT OF LINE
43015   11002        ISZ     PLC.,2     ;STEP OVER LINE #
43016   11002        ISZ     PLC.,2     ;STEP OVER LINE ADDRESS
43017    6331 PLNR2: NEXTBYTE           ;BEGIN SCANNING A STATEMENT
43020   20223        LDA     0,C125
43021  112655        SUBOR#  0,2,SNR    ;DATA OR REM STATEMENT ?
43022     762        JMP     PLNR1      ;  YES, GOTO NEXT LINE
43023   20215        LDA     0,C101     ;  NO
43024  142655        SUBOR#  2,0,SNR    ;"GOTO" OR "GOSUB" ?
43025     430        JMP     PLNR3      ;  YES
43026   20752        LDA     0,C140
43027  112655        SUBOR#  0,2,SNR    ;IF jump or ELSE statement?
43030     446        JMP     SGG1       ;  Yes, Skip over like GOTO
43031   50545        STA     2,IFON     ;  NO, SET FLAG TO TOKEN CODE
43032    6331 PLNR4: NEXTBYTE           ;SCAN STATEMENT
43033   24251        LDA     1,C342
43034  132015        ADC#    1,2,SNR    ;NEW STATEMENT FOLLOW ?
43035     412        JMP     PLNR6      ;  YES
43036  146415        SNE     2,1        ;END OF STATEMENT ?
43037     745        JMP     PLNR1      ;  YES
43040   24255        LDA     1,C351     ;  NO
43041  146414        SEQ     2,1        ;LITERAL STRING ?
43042     447        JMP     PLNR5      ;  NO
43043    6331        NEXTBYTE           ;  YES
43044  151014        SKZ     2,2        ;SCAN TO END OF STRING
43045     776        JMP     .-2
43046     764        JMP     PLNR4      ;CONTINUE SCANNING STATEMENT

43047   34355 PLNR6: LDA     3,.BUS     ;NEW STATEMENT
43050   31403        LDA     2,PBC.,3
43051  151620        INCZR   2,2        ;SET PBC TO EVEN BYTE NUMBER
43052  151120        MOVZL   2,2
43053   51403        STA     2,PBC.,3
43054     743        JMP     PLNR2      ;SCAN NEW STATEMENT

43055   26520 PLNR3: LDA     1,@.LPO    ;LIST-PROTECT OPTIONS
43056  124005        COM     1,1,SNR    ;  1 ?
43057     404        JMP     SGG        ;  YES, SWAP
43060  124202        COMR    1,1,SZC    ;  3 ?
43061     402        JMP     SGG        ;  YES, SWAP
43062     414        JMP     SGG1       ;  NO, LEAVE AS IS
```

```
;                    << SI = R94SPP4S7B; BO = 1/A.SPP4.9062! >>

43063  34355 SGG:   LDA     3,.BUS
43064  15403        DSZ     PBC.,3
43065 126520        SUBZL   1,1         ; +1
43066 151212        SKE     2,2         ; 101 ?
43067 126000        ADC     1,1         ;  YES, CONVERT TO 100
43070 133000        ADD     1,2         ;CONVERT
43071 141000        MOV     2,0
43072  25403        LDA     1,PBC.,3
43073   6145        JSR     @XPUTB&377;CHANGE BASIC STATEMENT
43074  34355        LDA     3,.BUS
43075  11403        ISZ     PBC.,3
43076   6331 SGG1:  NEXTBYTE            ;GET NUMBER OF LINE'S
43077 151120        MOVZL   2,2         ;WORDS TO BYTES
43100  34355        LDA     3,.BUS
43101  25403        LDA     1,PBC.,3
43102 147000        ADD     2,1         ;UPDATE POINTER
43103  45403        STA     1,PBC.,3
43104   6331        NEXTBYTE
43105  20251        LDA     0,C342
43106 112015        ADC#    0,2,SNR
43107    740        JMP     PLNR6
43110    674        JMP     PLNR1

43111  24226 PLNR5: LDA     1,C174     ;DECIMAL NUMBER ?
43112  20053        LDA     0,C200
43113 142032        SGE     2,0
43114 146022        ADCZ    2,1,SZC
43115    407        JMP     PLNR9      ;  NO
43116  30355 PLNRC: LDA     2,.BUS     ;  YES
43117  21003        LDA     0,PBC.,2
43120 101620        INCZR   0,0         ;SET PBC BEYOND NUMBER
43121 122520        SUBZL   1,0
43122  41003        STA     0,PBC.,2
43123    707        JMP     PLNR4      ;CONTINUE SCANNING

43124  24032 PLNR9: LDA     1,C12
43125  20035        LDA     0,C15
43126 142032        SGE     2,0         ;SUBROUTINE NUMBER ?
43127 146022        ADCZ    2,1,SZC
43130    402        SKIP                ;  NO
43131    765        JMP     PLNRC      ;  YES
43132  20270        LDA     0,C375
43133 112414        SEQ     0,2         ;TERMINAL MNEMONIC ?
43134    406        JMP     PLNR7      ;  NO
43135   6331 PLNRA: NEXTBYTE            ;  YES, SKIP STRING
43136  20375        LDA     0,375
43137 112414        SEQ     0,2
43140    775        JMP     PLNRA
43141    671        JMP     PLNR4      ;NEXT TOKEN
```

```
;                       << SI = R94SPP4S7B; BO = 1/A.SPP4.9062! >>

43142  20434 PLNR7: LDA     0,IFON
43143  24220        LDA     1,C114
43144 106415        SNE     0,1        ;'IF' ?
43145    404        JMP     PLN70      ;  Yes, Check options
43146  24225        LDA     1,C136
43147 122014        ADC#    1,0,SZR    ;MultistmtIF (IF/ENDIF)?
43150    662        JMP     PLNR4      ;  NO
43151  22424 PLN70: LDA     0,@.LPO    ;  YES, LIST-PROTECT OPTIONS
43152 101015        SNZ     0,0        ;2 ?
43153    403        JMP     PLNR8      ;  YES, CONVERT RELATIONAL OPERATOR
43154 100005        COM     0,0,SNR    ;3 ?
43155    655        JMP     PLNR4      ;  NO, CONTINUE SCAN
43156  20260 PLNR8: LDA     0,C357     ;  YES
43157 142433        SLE     2,0        ;RELATIONAL OPERATOR ?
43160    652        JMP     PLNR4      ;  NO, CONTINUE SCAN
43161  20256        LDA     0,C352     ;  MAYBE...
43162 142032        SGE     2,0
43163    647        JMP     PLNR4      ;  NO, CONTINUE SCAN
43164 141220        MOVZR   2,0        ;  YES, TOGGLE LSB
43165 101160        MOVCL   0,0
43166  34355        LDA     3,.BUS
43167  15403        DSZ     PBC.,3
43170  25403        LDA     1,PBC.,3
43171   6145        JSR     @XPUTB&377;CONVERT OPERATOR
43172  34355        LDA     3,.BUS
43173  11403        ISZ     PBC.,3
43174    636        JMP     PLNR4      ;CONTINUE SCAN

43175  42515 .LPO:  LPO
43176      0 IFON:  0
```

```
;                        << SI = R94SPP4S7B; BO = 1/A.SPP4.9062! >>

;---------------------------------------------------------------
;
;  THE FOLLOWING ROUTINE CHECKS TO SEE THAT A VALID R8 PICO-N
; IS ON THE SYSTEM, & THAT A SPECIFIC LEVEL 2 KEY ASSIGNED
; TO THIS PROCESSOR, IS TURNED ON.
; IF THESE CHECKS FAIL, THE POINT 4 PROTECTION WILL NOT BE
; MADE TO THE BASIC PROGRAM, HOWEVER THE OPERATOR WILL NOT
; BE MADE AWARE OF THIS.
;
;---------------------------------------------------------------

    43177    4440 PNC:   JSR     PNC1
    43200      36        .BLK   36          ;WORK SPACE
    43236    1731 PNK:   1731    ;LEVEL 2 KEY ASSIGNED TO SPP4 (985 DECIMAL)

    43237   20777 PNC1:  LDA     0,PNK
    43240  165000        MOV     3,1
    43241    4405        JSR     PND2       ;VALIDATE P-N & CHECK KEY
    43242    2403        JMP     @.P4K      ;OK TO RUN SPP4
    43243    2401        JMP     @.+1       ;   BAD
    43244   34051        RUNC1
    43245   42516 .P4K:  P4K
```

```
;                        << SI = R94SPP4S7B; BO = 1/A.SPP4.9062! >>
;-------------------------------------------------------------------
;
;
;PICO-N DRIVER #2

; ENTER WITH INTERRUPTS ENABLED
; A0 = 10-BIT NUMBER TO BE VERIFIED
;       BITS 0-8 = P-N ADDRESS USED IN CONV (A0).
;       BIT 9 = BUSY-DONE INDICATOR (0=B, 1=D).
;             FOR EXAMPLE, TO CHECK B OR D BIT IN P-N
;             LOCATION 324 (ROW 32, COL 4), A0 MUST CONTAIN:
;                0423 TO CHECK BUSY
;                1423 TO CHECK DONE
; A1 POINTING TO WORKSPACE (36 WORDS)
; SKIP RETURN IF B-D BIT = 0, OR IF P-N FAILED VALIDATION.
;
;-------------------------------------------------------------------
```

```
;                    << SI = R94SPP4S7B;  BO = 1/A.SPP4.9062! >>

 43246  54535 PND2:  STA     3,PTN
 43247  44535        STA     1,PNTR
 43250  44535        STA     1,PNTRO
 43251  40535        STA     0,S0
 43252  22460        LDA     0,@.TSC   ;VARIABLE SEQUENCE
 43253  24460        LDA     1,C1400   ;BUILD UP TO 4 INSTRUCTIONS TO
 43254 107700        ANDS    0,1       ;MISLEAD THE OBSERVER.
 43255 124000        COM     1,1
 43256   4460        JSR     CONV
 43257 125404        INC     1,1,SZR
 43260    776        JMP     .-2
 43261  24445        LDA     1,REST    ;RESET THE PICO-N
 43262 125300        MOVS    1,1       ;DIC 0,40
 43263  46521        STA     1,@PNTR
 43264  10520        ISZ     PNTR
 43265  20442        LDA     0,VAL1    ;STEP THRU THE
 43266  24447        LDA     1,CM4     ;FIRST 4 ROWS OF THE PICO-N.
 43267   4447        JSR     CONV      ;GENERATE 1 INSTRUCTION.
 43270 125404        INC     1,1,SZR
 43271    776        JMP     .-2
 43272  20436        LDA     0,VAL2    ;4 MORE STEPS W. VALIDATION (ROWS 4-7)
 43273  30436        LDA     2,VAL3
 43274  24441        LDA     1,CM4
 43275   4441        JSR     CONV
 43276   4461        JSR     CKBD      ;GEN 2 INSTR. TO VAL. B & D BITS.
 43277 125404        INC     1,1,SZR
 43300    775        JMP     .-3
 43301  20505        LDA     0,S0      ;STEP XYZ (9 BITS OF GIVEN CODE)
 43302   4434        JSR     CONV
 43303   4433        JSR     CONV
 43304   4432        JSR     CONV
 43305  24475        LDA     1,SKBZ    ;TEST B OR D BIT
 43306 125300        MOVS    1,1       ;DECODE
 43307  30053        LDA     2,C200
 43310 101212        SKE     0,0       ;TEST BUSY BIT ??
 43311 147000        ADD     2,1       ;   NO, GEN SKPDZ INSTR.
 43312  46472        STA     1,@PNTR
 43313  30471        LDA     2,PNTR    ;SKIP OR NON-SKIP RETURN
 43314  20417        LDA     0,J03
 43315  41001        STA     0,1,2
 43316 101400        INC     0,0
 43317  41002        STA     0,2,2
 43320  60277        INTDS
 43321   6464        JSR     @PNTRO    ;EXECUTE PREPARED CODE (IN WORK SPACE)
 43322    402        SKIP              ;SUCCESSFUL RETURN
 43323  10460        ISZ     PTN       ;BAD RETURN - BIT = 0, OR VALIDATION FAILED.
 43324  60177        INTEN
 43325   2456        JMP     @PTN
```

```
;                        << SI = R94SPP4S7B;  BO = 1/A.SPP4.9062!  >>
    43326   20145 REST:  020145 ;DIC 0,40   SWAPPED
    43327    3051 VAL1:  3051   ;ROWS 0-3
    43330    2340 VAL2:  2340   ;ROWS 4-7
    43331     306 VAL3:  3*4+0*4+1*4+2    ;B&D VALUES FOR ROWS 7-4 = 11,00,01,10
    43332     637 .TSC:  INFO+.TSC.
    43333    1400 C1400: 1400
    43334  177773 CM5:   -5
    43335  177774 CM4:   -4

;  CONVERT AO (0-2) TO INSTR & STORE
;       BIT 0 -    0=DIC,  1=DIB
;       BIT 1 -    0=DEVICE CODE X5,  1=X4
;       BIT 2 -    0=DEVICE CODE 7X,  1=2X

;      AO    INSTR.  GENERATED
;     ----   ----------------
;      0        DIC    0,75
;      1        DIB    0,75
;      2        DIC    0,74
;      3        DIB    0,74
;      4        DIC    0,25
;      5        DIB    0,25
;      6        DIC    0,24
;      7        DIB    0,24

    43336   54451 CONV:  STA     3,PTN2
    43337   44451        STA     1,R1
    43340   24416        LDA     1,BASE
    43341  125300        MOVS    1,1
    43342   34067        LDA     3,C1000
    43343  101223        MOVZR   0,0,SNC   ;DIB ??
    43344  167000        ADD     3,1       ;   NO, GEN DIC
    43345  101223        MOVZR   0,0,SNC   ;DEVICE CODE X4 ??
    43346  125400        INC     1,1       ;   NO, GEN DEV CODE X5
    43347   34442        LDA     3,C50
    43350  101223        MOVZR   0,0,SNC   ;DEVICE CODE 2X ??
    43351  167000        ADD     3,1       ;   NO, GEN DEV CODE 7X
    43352   46432 SRET:  STA     1,@PNTR
    43353   10431        ISZ     PNTR
    43354   24434        LDA     1,R1
    43355    2432        JMP     @PTN2

    43356   12143 BASE:  12143             ; DIB  0,24  SWAPPED
```

```
;                        << SI = R94SPP4S7B;  BO = 1/A.SPP4.9062! >>
;USED FOR VALIDATION OF ROWS 4-7 ONLY.
;GENERATES 3 INSTRUCTIONS TO CHECK B & D BITS:
;  ON INPUT:      A2 BIT 0 = DONE BIT SETTING
;                    BIT 1 = BUSY BIT SETTING

;  INSTRUCTIONS GENERATED:
;       SKPDZ/N  0
;       SKPBZ/N  0
;       JMP      1,3


 43357  54430 CKBD:  STA     3,PTN2
 43360  44430        STA     1,R1
 43361  34051        LDA     3,C100
 43362  24417        LDA     1,SKDN
 43363 125300        MOVS    1,1         ;DECODE
 43364 151222        MOVZR   2,2,SZC     ;BIT 0 = 0 ??
 43365 167000        ADD     3,1         ;   NO, GEN SKPDZ
 43366  46416        STA     1,@PNTR
 43367  10415        ISZ     PNTR
 43370  24412        LDA     1,SKBZ
 43371 125300        MOVS    1,1
 43372 151222        MOVZR   2,2,SZC     ;BIT 1 = 0 ??
 43373 166400        SUB     3,1         ;   NO, GEN SKPBN
 43374  46410        STA     1,@PNTR     ;SAVE
 43375  10407        ISZ     PNTR
 43376  24735        LDA     1,J03       ;GEN JMP 1,3
 43377 125400        INC     1,1
 43400    752        JMP     SRET

 43401 100147 SKDN:  100147  ;63600 SWAPPED
 43402  40147 SKBZ:  40147   ;63500 SWAPPED
        43333 J03=   C1400
 43403      0 PTN:   0       ;RETURN ADDR
 43404      0 PNTR:  0
 43405      0 PNTR0: 0
 43406      0 S0:    0
 43407      0 PTN2:  0
 43410      0 R1:    0
 43411     50 C50:   50

              .EOT   ;  'PROTECT' R8.2 SOURCE #7
```

<< SI = R94BASSKPSB; BO = 1/A.SPP4.9062! >>

```
;                    << SI = R94BASSKPSB; BO = 1/A.SPP4.9062! >>

;
;     R9XSKPEXP  -  Common source file included in the BASIC processor
;                   and the RESOLVE discsub.
;
;     27-Feb-89,   nm: Check EndOfPGM when skip to next statement
;
;     SLSTM  -  Set current position of NXSTM to specified line address
;               which may or may not be in partition and then return
;               the address and the code of the first statement.
;
;     Calling Sequence:
;
;                       ;Set SBA as necessary
;       LDA   2,SBARelativeAddressOfLine
;       JSR   SLSTM
;       JMP   EndOfPGM ;Impossible
;       STA   0,OpcodeOfFirstStatement
;       STA   2,AddressOfFirstStatement
;
;     FRSTM  -  Set current position of NXSTM routine to start of
;               IRIS BASIC program in partition and then return the
;               address and the code of the first statement.
;
;     Calling Sequence:
;
;       JSR   FRSTM
;       JMP   EndOfPgm ;Program area is empty
;       STA   0,OpcodeOfFirstStatement
;       STA   1,RelAddressOfSLTentryForStmt
;       STA   2,AddressOfFirstStatement
;
;     NXSTM  -  Return address and opcode of next IRIS BASIC statement.
;
;     Calling Sequence:
;
;                       ;Initialize current position by calling FRSTM
;
;       JSR   NXSTM
;       JMP   EndOfPgm
;       STA   0,OpcodeOfStatement
;       STA   1,RelAddressOfSLTEntryForStmt
;       STA   2,AddressOfStatement
;
```

```
;                    << SI = R94BASSKPSB; BO = 1/A.SPP4.9062! >>


43412  54414 SLSTM: STA    3,NXSRT   ;Save return address
43413  34023        LDA    3,CM400
43414  54413        STA    3,NXSLO   ;Set PLC index to END
43415    472        JMP    NXSFS     ;Set NXSTM and return first stmt info


43416  54410 FRSTM: STA    3,NXSRT   ;Save return address
43417  36004        LDA    3,@PIB    ;Load partition address
43420  31405        LDA    2,SLT.,3  ;Load rel address of first stmt entry in SLT
43421    510        JMP    NXSFL     ;And setup first line

43422    125 VNS125:  125
43423    173 VNS173:  173
43424     73 VNS73: 73
43425    140 VNS140:  140

43426      0 NXSRT: 0              ;Return address for FRSTM and NXSTM
43427      0 NXSLO: 0              ;Rel address of current line in statement tbl (SLT)
43430      0 NXSSA: 0              ;Relative address of current statement
43431      0 NXSOP: 0              ;Opcode of current statement
```

```
;                    << SI = R94BASSKPSB; BO = 1/A.SPP4.9062! >>

 43432  54774 NXSTM: STA     3,NXSRT   ;Save return address

 43433  30775        LDA     2,NXSSA   ;Load current statement rel address
 43434  20775        LDA     0,NXSOP   ;Load current statement opcode
 43435  24051        LDA     1,C100
 43436 122655        SUBOR#  1,0,SNR   ;GOTO or GOSUB?
 43437    523        JMP     NXSGO     ;  Yes, Skip over line numbers
 43440  24762        LDA     1,VNS125
 43441 122655        SUBOR#  1,0,SNR   ;DATA or REM?
 43442    463        JMP     NXSNL     ;  Yes, Goto to next line
 43443  24762        LDA     1,VNS140
 43444 122655        SUBOR#  1,0,SNR   ;MultiStmtIF branch or ELSE?
 43445    515        JMP     NXSGO     ;  Yes, Treat as GOTO
 43446 145140        MOVOL   2,1       ;Increment past opcode and start skipping over stmt

                ;    Skip over next token if not end of statement
                ;    A1 = byte address of next token

 43447   6144 NXSNX: XGETBYTE          ;Load first byte of token
 43450 125400        INC     1,1       ;Increment byte address
 43451  20053        LDA     0,C200    ;Token over 200 octal?
 43452 142033        SLS     2,0
 43453    415        JMP     NXSCV     ;  Yes
 43454  20747        LDA     0,VNS173
 43455 142433        SLE     2,0       ;Floating point constant?
 43456    465        JMP     NXSFC     ;  Yes, Skip over it
 43457  20745        LDA     0,VNS73
 43460 142654        SUBOR#  2,0,SZR   ;Extended variable number?
 43461 112255        ADCOR#  0,2,SNR
 43462    456        JMP     NXSS1     ;  Yes, Skip next byte
 43463  20032        LDA     0,C12
 43464 112414        SEQ     0,2       ;"CALL" subroutine name?
 43465 112255        ADCOR#  0,2,SNR
 43466    454        JMP     NXSNC     ;  Yes, Skip over it
 43467    760        JMP     NXSNX     ;Single byte token, Try next
```

```
;                    << SI = R94BASSKPSB; BO = 1/A.SPP4.9062! >>

                ;     Token in A2 is greater than or equal to 200, A1 = next byte address

43470  20431 NXSCV: LDA     0,VNSMV   ;Variable?
43471 142032         SGE     2,0
43472    755         JMP     NXSNX     ;  Yes, Try next token
43473  20430         LDA     0,VNS351  ;String literal ("xxx")?
43474 142415         SNE     2,0
43475    452         JMP     NXSSL     ;  Yes, Skip over it
43476  20426         LDA     0,VNS375  ;Mnemonic string ('xxxx')?
43477 142415         SNE     2,0
43500    454         JMP     NXSMS     ;  Yes, Skip over it
43501  20421         LDA     0,VNS342
43502 112654         SUBOR#  0,2,SZR   ;End of statement?
43503    744         JMP     NXSNX     ;  No, Try next token
43504 112415         SNE     0,2       ;End of line?
43505    420         JMP     NXSNL     ;  Yes, Skip to next line
43506 131620         INCZR   1,2       ;Next Statement code, Roundup to word boundary

                ;     Found next statement, A2 = relative address of stmt

43507  50721 NXSFS: STA     2,NXSSA   ;Set new current statement address
43510  34040         LDA     3,SBA     ;Load base address
43511 173000         ADD     3,2       ;Calculate absolute address of statement
43512  21000         LDA     0,0,2     ;Load first word of statement
43513  24023         LDA     1,CM400
43514 123700         ANDS    1,0       ;Extract opcode of statement
43515  40714         STA     0,NXSOP   ;Save opcode
43516  24711         LDA     1,NXSLO   ;Return SLT offset in A1
43517  34707         LDA     3,NXSRT   ;Perform skip return
43520   1401         JMP     1,3

43521    340 VNSMV:  340
43522    342 VNS342:  342
43523    351 VNS351:  351
43524    375 VNS375:  375

                ;     Skip to next line

43525  36004 NXSNL: LDA     3,@PIB    ;Load address of partition
43526  30701         LDA     2,NXSLO   ;Load offset of current line in stmt tbl (SLT)
43527 151400         INC     2,2       ;Add two to advance to next line
43530 151400         INC     2,2
43531  21401 NXSFL: LDA     0,VDT.,3  ;Load address of variable tbl (end of SLT)
43532 142033         SLS     2,0       ;End of program?
43533   2673         JMP     @NXSRT    ;  Yes, Perform nonskip return
43534  50673         STA     2,NXSLO   ;  No, Set new current line offset
43535 173000         ADD     3,2       ;Calculate abs address of line
43536  31001         LDA     2,1,2     ;Load relative address of line
43537    750         JMP     NXSFS     ;Return with statement address and opcode

                ;     Skip next byte and continue

43540 125400 NXSS1: INC     1,1       ;Increment byte address
43541    706         JMP     NXSNX     ;And try next token
```

```
;                    << SI = R94BASSKPSB; BO = 1/A.SPP4.9062! >>

                ;     Skip over "CALL" subroutine name, A0 = 12

43542 151400 NXSNC: INC     2,2       ;Increment token value and then handle as Flt Pnt const

                ;     Skip over multi-word constant, A0 = base of token range - 1

43543 112400 NXSFC: SUB     0,2       ;Calculate number of words in token
43544 125620        INCZR   1,1       ;Roundup byte pointer to word pointer
43545 147120        ADDZL   2,1       ;Skip over constant and convert back to byte pointer
43546    701        JMP     NXSNX     ;And try next token

                ;     Skip to end of string literal ("xxxx")

43547   6144 NXSSL: XGETBYTE          ;Load byte from string
43550 125400        INC     1,1       ;Increment pointer to next byte
43551 151014        SKZ     2,2       ;End of string?
43552    775        JMP     NXSSL     ;   No, Continue skipping
43553    674        JMP     NXSNX     ;   Yes, Try next token

                ;     Skip to end of mnemonic string ('xxxx')

43554   6144 NXSMS: XGETBYTE          ;Load byte from string
43555 125400        INC     1,1       ;Increment pointer to next byte
43556  20746        LDA     0,VNS375
43557 142414        SEQ     2,0       ;Closing single quote?
43560    774        JMP     NXSMS     ;   No, Continue skipping
43561    666        JMP     NXSNX     ;   Yes, Try next token

                ;     Skip over GOTO type statement.  Second byte of statement
                ;     specifies the number of words to skip.  These words (line#s
                ;     in GOTO) may be followed by a 343, NextStatement, code.  If
                ;     so, the next statement is in the same line at the next word
                ;     boundary.  Otherwise, the next statement is the first statement
                ;     of the next line.

43562  34040 NXSGO: LDA     3,SBA     ;Load base address
43563 173000        ADD     3,2       ;Calculate abs address of statement
43564  21000        LDA     0,0,2     ;Load first word of statement
43565  24064        LDA     1,C377
43566 123400        AND     1,0       ;Extract number of words to skip
43567 113000        ADD     0,2       ;And skip them
43570  21001        LDA     0,1,2     ;Load next word of statement
43571 101300        MOVS    0,0       ;Extract next byte
43572 123400        AND     1,0
43573  24727        LDA     1,VNS342
43574 122014        ADC#    1,0,SZR   ;Next Statement code?
43575    730        JMP     NXSNL     ;   No, Skip to next line
43576 151400        INC     2,2       ;   Yes, Skip word to next statement
43577 151400        INC     2,2
43600 172400        SUB     3,2       ;Convert statement address to SBA relative
43601  36004        LDA     3,@PIB
43602  35406        LDA     3,UVS.,3  ;Load offset of pgm end
43603 172015        ADC#    3,2,SNR   ;Walked off end of pgm?
43604    721        JMP     NXSNL     ;   Yes, NextStmt code was junk
43605    702        JMP     NXSFS     ;And return statement found
```

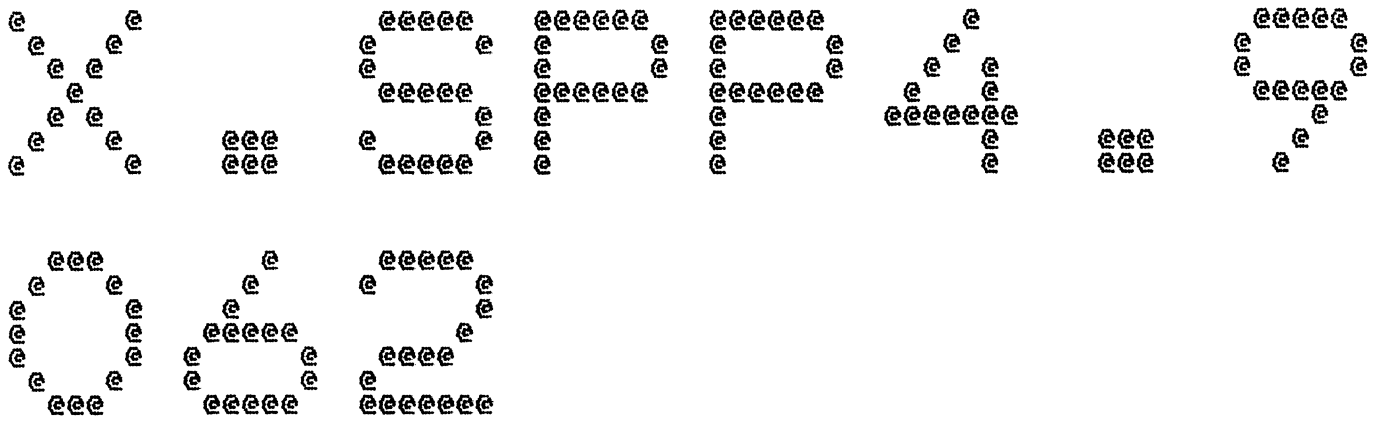
```
---

.EOT    ;R9XBASSKP
```

```
. END
```

A80	32267	A94	32271	ACNB	32644	ACNL	33127	ADK	42620
AE	100210	AF	36402	ASLT	42675	BADK	36235	BASE	43356
BBCNT	36666	BI	36202	BINDI	6115	BINMU	6116	BIPPY	34025
BLDFL	36655	BPI	16	BRDEF	200	BSACF	75	BSCRV	134664
BUMPU	6117	C10	30	C100	51	C1000	67	C101	215
C105	216	C106	36412	C11	31	C112	217	C114	220
C115	221	C12	32	C121	222	C125	223	C13	33
C132	224	C136	225	C14	34	C140	43000	C1400	43333
C15	35	C16	36	C160	174	C163	175	C166	176
C17	37	C170K	21	C171	177	C174	226	C177	52
C1777	70	C2	2	C20	42	C200	53	C2000	71
C205	54	C212	227	C215	55	C22	211	C232	230
C240	56	C242	231	C244	57	C245	232	C247	233
C250	234	C251	235	C254	236	C256	237	C260	60
C262	42507	C271	61	C272	240	C273	241	C275	242
C276	243	C3	3	C300	62	C32	212	C331	244
C333	245	C334	63	C335	246	C336	247	C340	250
C342	251	C343	252	C344	253	C350	254	C351	255
C352	256	C353	257	C357	260	C361	261	C362	262
C364	263	C365	264	C366	265	C37	43	C370	266
C374	267	C375	270	C376	271	C377	64	C4	24
C40	44	C400	65	C4000	72	C5	25	C50	43411
C6	26	C60	213	C600	100	C67	36413	C7	27
C71	214	C77	50	C774C	22	C777	66	CALCK	36340
CALL	6101	CBO	36354	CCAO	32722	CCDON	32454	CCFLG	32460
CHANN	6106	CKBD	43357	CKEY	36203	CKEY1	36212	CLMSK	32225
CLNO	42663	CM10	36502	CM4	43335	CM400	23	CM5	43334
CM7	36501	CO	36370	COMB	33133	CONV	43336	COPYP	32730
CPYP2	32746	CR	272	CRTL	35651	CUK	36254	CVSP	32345
CVSP1	32425	CVSP2	32456	DA	160	DAC	164	DAS	165
DATAP	6110	DB	166	DBA	41	DBC	172	DBS	173
DCD	37670	DCDON	37711	DCOD	37041	DECIM	6120	DEC.	21
DET.	25	DFD	32605	DFS.	31	DFTCA	34106	DLTE	33425
DLTE1	33443	DMCAL	34110	DQUEU	6105	DSC.	22	DSQ	37666
DUMP	36756	DWC.	20	D.SAV	200	EBP	353	EBS	41200
ECL1	42721	ECL2	42732	ECR2	42707	EDEX	34336	EDEX1	34406
EDEX2	34326	EDEX3	34412	EDEX5	34601	EDEX7	35021	EDEX8	35050
EDEXB	35140	EDEXC	35156	EDEXD	34630	EDEXE	34640	EDEXF	34725
EDEXG	34576	EDEXK	34257	EDEXL	34701	EDEXN	34767	EDEXO	34777
EDEXQ	35063	EDEXR	34717	EDEXS	34515	EDEXT	34763	EDEXV	34460
EDEXW	34521	EDEXX	35057	EDEXZ	35056	EDITC	33770	EDITI	33671
EDITK	33645	EDITL	33273	EDITO	33416	EDITR	33261	EDITS	33400
EDM8A	35622	EDML	35176	EDML1	35447	EDML2	35461	EDML3	35504
EDML4	35527	EDML5	35545	EDML6	35555	EDML7	35575	EDML8	35613
EDML9	35631	EDMLA	35313	EDMLB	35324	EDMLD	35257	EDMLI	35242
EDMLN	35410	EDTBL	36115	EDUN	33500	EDX3A	34436	EDX3B	34446
EDXEE	34666	EDXK1	34275	EDXK2	34307	EDXO1	35006	EDXQ1	35114
EDXQ2	35131	EDXRX	34723	EDXW1	34546	EDXW2	34570	EICB	33317
EL	44000	EL2	43730	ELSSZ	50	EMDON	37662	EMN	37642
EMOD1	20	EN	44050	ENC1	36461	ENCD2	42717	ENCDE	36430
ENCR	36415	ENT1	33531	EP6	42514	ERBP.	52	ERLN.	53
ERR	32612	ERR1	32637	ERR2	32641	ERRF	76	ERRLN	204
ERRN.	51	ERRSO	202	ESCF	73	ESQ	37640	ETSF	74
EUS.	10	F4MOD	40	F4TRA	10	FDEFS	200	FESIZ	14
FIND	37715	FINDL	6123	FIVRD	0	FIX	6121	FL1.	54
FL2.	56	FL3.	57	FL4.	60	FLAGC	6102	FLAG.	24
FLNCS	36020	FLNFR	36035	FLNNL	36031	FLNNS	36016	FLNR	36004
FLOAT	6122	FNLIM	2	FNPBC	6	FNPLC	1	FNS	177

FNSPR	13	FNSS	10	FNSTP	7	FNS.	14	FOUND	37727
FREEN	6107	FRSTM	43416	FSC.	15	FSLN	37513	GETBY	6124
GSC.	17	GSS	337	GSSS	10	GSS.	16	HELP	32774
HELPC	33027	IFON	43176	IF.	34120	INBYT	6125	INLEN	64
INSTB	6126	IOCAL	34103	IOP	6	ISA2D	6127	ISA2L	6130
JO3	43333	JFLTO	151	JLSS1	37431	JLSTE	37346	JLSTS	37400
KEYC	36262	L1KEY	36426	L2KEY	42715	LBS	343	LIST	37013
LIST1	37021	LISTA	37327	LISTC	37602	LISTD	37432	LISTE	37504
LISTF	37574	LISTG	37272	LISTI	37571	LISTK	37545	LISTL	37053
LISTO	37477	LISTR	37231	LISTS	37261	LISTT	36700	LISTU	37532
LISTV	37373	LISTW	37543	LISTX	37152	LISTZ	37353	LLDR	36722
LLDRX	36747	LMAP	35655	LNO	346	LOAD	33470	LOADD	6131
LOOK	37717	LPMSK	36001	LPO	42515	LREV	41300	LRTC1	36564
LSTA1	37337	LSTBL	36123	LSTC1	37633	LSTC2	37610	LSTD1	37455
LSTG1	37306	LSTG2	37321	LSTL1	37123	LSTL2	37137	LSTR1	37255
LSTRS	37252	LSTTO	36715	LSTX1	37212	LWICB	36616	LWRRL	36154
LWRTC	36546	LWRTE	36531	LWRTX	36610	MFLAG	273	MLPM	36002
MPYR	36176	MXPRC	4	N9999	317	NC1	33551	NCOD	33532
NCOD1	33647	NCOD2	33715	NCOD3	33736	NCOD4	34011	NCODD	33675
NCODE	6330	NEW	32517	NEXF	351	NEXTB	6331	NOKEY	42531
NTES	347	NVLP	42544	NVS.	7	NXSCV	43470	NXSFC	43543
NXSFL	43531	NXSFS	43507	NXSGO	43562	NXSLO	43427	NXSMS	43554
NXSNC	43542	NXSNL	43525	NXSNX	43447	NXSOP	43431	NXSRT	43426
NXSS1	43540	NXSSA	43430	NXSSL	43547	NXSTM	43432	OKEY	36252
ONF	344	OPLD	36620	OPLDD	36626	OPLDU	36642	OPLDX	36647
OTMS	32245	OUTBY	6132	OUTTE	6133	P	200	P4K	42516
P4M	42606	PBC.	3	PIB	4	PKEY	36316	PLC.	2
PLN70	43151	PLNR	43001	PLNR1	43004	PLNR2	43017	PLNR3	43055
PLNR4	43032	PLNR5	43111	PLNR6	43047	PLNR7	43142	PLNR8	43156
PLNR9	43124	PLNRA	43135	PLNRC	43116	PLTBL	42200	PMSK	36003
PNC	43177	PNC1	43237	PND2	43246	PNK	43236	PNTR	43404
PNTRO	43405	POKE	35726	POKEY	35777	PRN	42704	PSTS.	55
PTN	43403	PTN2	43407	PUTBY	6134	QCHAR	6103	QFLAG	33463
QUEUE	6104	R1	43410	R8FIX	32226	R8FX2	32236	RC1	36263
RDKY	36324	READB	6135	RELJM	6136	RELO1	36173	RES	447
REST	43326	REV.	0	RFBA	354	RFW	36000	RIBP	36312
RJSR	6136	RKEY	36251	RNDI.	45	RNDM	36503	RNDM1	42665
RNR	36525	RNUM	35664	RNUM1	35672	RNUM2	35665	RNUM3	35706
RNUM4	35712	ROBP.	23	RTN	36427	RTN1	36530	RTN2	42716
RTP	7	RUNC	34027	RUNC1	34051	RUP	5	SO	43406
SA1	36526	SA2	36527	SATUP	420	SAVEB	6332	SBA	40
SCDCA	34147	SE	100010	SENT	33512	SENTC	33344	SENTE	33306
SFN	36301	SFSN	32751	SFSN1	32757	SFTYP	451	SFVN	33143
SFVN1	33154	SFVN2	33174	SFVN3	33253	SGG	43063	SGG1	43076
SIBP	36264	SIZE	32650	SKBZ	43402	SKDN	43401	SKEY	36265
SKEY1	36321	SLPB	42602	SLSTM	43412	SLT	457	SLT.	5
SNI	32473	SNIO	32467	SNI1	32472	SNI2	32471	SNI3	32470
SNIF	32477	SNIP	32505	SNO	32461	SNOP	32463	SNOR	32464
SNTCX	33327	SNTRS	33335	SPINP	6146	SPP4	42501	SRET	43352
SSC	345	STEB	32713	STINP	6140	STINT	6147	STORD	6137
STOUT	6141	STPBC	63	STPRC	24	STUP1	427	STYPE	352
SWPI	377	SWPI1	405	SWPO	32274	SWPO1	32337	SWPO2	32342
SWTB	33051	SWTBC	33123	SWTBD	33117	TAPE	36753	TASKQ	15
TCH	42501	TRAPF	6142	TRP1	32217	TSE	333	TSE1	334
TSE2	335	TSE3	336	TSE4	337	TSE5	340	TSE6	341
TSE7	342	TSNS	11	TSN.	103	TSU.	73	TSU. 1	74
TSU. 2	75	TSU. 3	76	TSU. 4	77	TSU. 5	100	TSU. 6	101
TSU. 7	102	UFC.	13	UFS	146	UFSS	5	UFS.	12

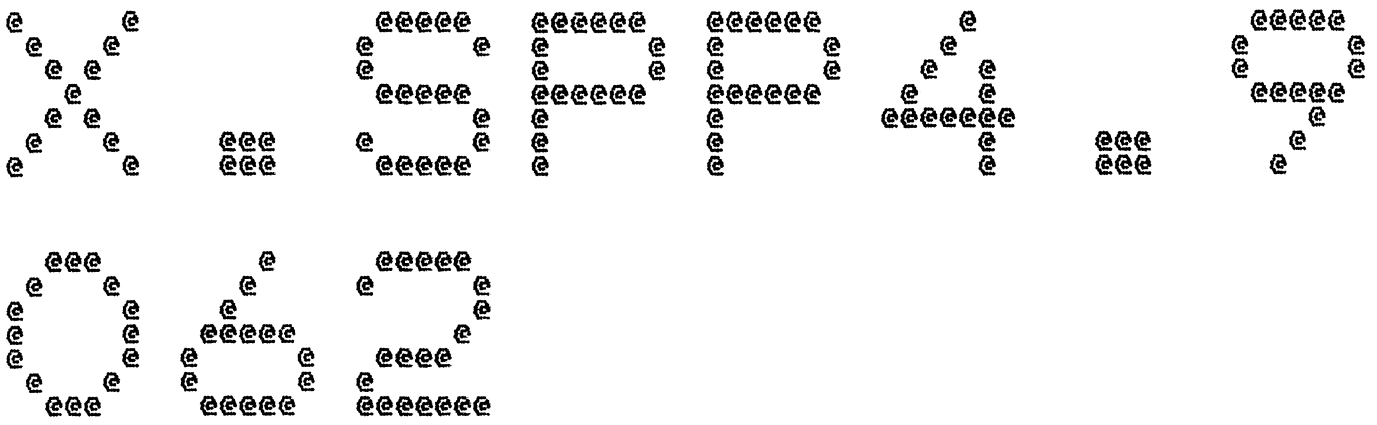
UFT	114	UFT.	11	UPS.	4	UVS.	6	VAL1	43327		
VAL2	43330	VAL3	43331	VDT.	1	VKOFF	42510	VLP	42570		
VNS12	43422	VNS14	43425	VNS17	43423	VNS34	43522	VNS35	43523		
VNS37	43524	VNS73	43424	VNSMV	43521	WORD	36042	WORD1	36057		
WORD2	36062	WRITB	6143	WTBL	34064	WTL	114	X9999	452		
XE	100	XGETB	6144	XOB	347	XOBE	62	XOBS	100		
XOB.	61	XPLNK	42644	XPLXK	42654	XPUTB	6145	.2NXS	36040		
.ABA	14	.ACNL	274	.BADK	42502	.BPS	77	.BSA	10		
.BUS	355	.CALK	42505	.CCAO	275	.CCDN	33465	.CCFG	33464		
.CD	37731	.CKEY	42511	.COMB	276	.CPYP	277	.CRTL	300		
.CUK	36253	.DA	174	.DA3	175	.DB	176	.DB3	177		
.DCOD	301	.DSQ	302	.EBS	206	.ECR2	42706	.EDEX	303		
.EDML	304	.EDTL	33663	.EDTO	33665	.EDTR	305	.EDTS	33664		
.EDUN	306	.EDXC	34520	.EDXE	307	.EDXF	34573	.EDXL	34250		
.EDXN	34574	.EDXO	34517	.EDXQ	310	.EDXR	34241	.EDXS	311		
.EDXT	34575	.EDXX	312	.ENC	36234	.ENCR	36414	.ESQ	313		
.ESS	207	.ESSE	210	.EUS	356	.FLTO	152	.FRST	36041		
.FSLN	36751	.HBA	11	.HXA	12	.INFO	100	.INTR	111		
.LCM	114	.LPO	43175	.LSTC	37263	.LSTD	37264	.LSTF	37265		
.LSTI	37266	.LSTK	37267	.LSTU	37270	.LSTW	37271	.LSTX	314		
.LW	36172	.LW5	36171	.LWR	36614	.MIDX	36174	.MN	37775		
.MPYR	36175	.MT	36173	.NCD1	33376	.NEW	316	.NRET	112		
.OKEY	42506	.P4K	43245	.PERM	317	.PLNR	42513	.PLTB	42512		
.R8FX	417	.RIBP	42503	.RNDM	42705	.RNUM	315	.SENT	320		
.SFSN	321	.SFVN	322	.SKEY	42504	.SNI	323	.SNO	324		
.SRET	113	.SSA	13	.STMT	350	.STYP	33666	.SWTB	325		
.TSC	43332	.WORD	326	.WTBL	327						



Spool Queue Line #: 2
IRIS LU/Filename : 1/X.SPP4.9062

Printed on/at : MAY 2, 1989 15:38:38
For Group/User: 2 , 2
On Port No: 36

Print control parameters :
Printer Class code : 0
Form Code/paper type : A
Print Priority (0-9) : 5
Starting Page Number : 1
This is copy number : 1
Keep file (Y/N) : N
Notify User when done: N
Comments, optional : For RDC



Spool Queue Line #: 2
IRIS LU/Filename : 1/X.SPP4.9062

Printed on/at : MAY 2, 1989 15:38:48
For Group/User: 2 , 2
On Port No: 36

Print control parameters :
Printer Class code : 0
Form Code/paper type : A
Print Priority (0-9) : 5
Starting Page Number : 1
This is copy number : 1
Keep file (Y/N) : N
Notify User when done: N
Comments, optional : For RDC

***** J O B S T A T I S T I C S *****

35	TOTAL # DUPLICATE KEYS
0	TOTAL # DIR. RE-ORGS
2,684	TOTAL # KEYS INSERTED
0	TOTAL # ASSEMBLY ERRS

```
. EUS       6.007:   6.030   13.016   15.017   20.049   48.037
. FRST     55.032   55.037:
. FSLN     69.053:  70.047   70.057
. HBA       7.006
. INFO      9.008   10.032   54.079   66.022
. LPO      93.055   95.014   95.035:
. LSTC     73.051:
. LSTD     72.021   73.052:
. LSTF     72.037   73.053:
. LSTI     72.031   73.054:
. LSTK     72.040   73.055:
. LSTU     72.034   73.056:
. LSTW     73.009   73.057:
. LSTX      4.022:  80.035
. LW       57.056   57.060:
. LW5      57.047   57.059:
. LWR      67.052   67.061:
. MIDX     59.025:  63.028   63.042   63.044
. MN       79.027   80.023   83.010:
. MPYR     59.026:  59.038
. MT       59.024:  63.018   63.027
. NCD1     21.054   23.034:  23.041   23.051   23.056
. NEW       4.024:  24.015
. OKEY     88.044:  89.010   89.016   90.020   90.039   90.053
. P4K      96.026   96.029:
. PERM      4.025=  22.044   57.050   68.033   68.059
```

C115	3.014:	23.047	23.055				
C12	15.022	15.037	29.024	35.030	77.026	94.042	104.038
C121	3.015:	42.019	67.013	72.024			
C125	3.016:	27.046	51.025	56.046	71.038	72.019	93.024
C132	3.017:	72.032					
C136	3.018:	20.025	23.044	95.011			
C14	39.020						
C140	93.007:	93.030					
C1400	98.012	99.011:	100.038				
C15	8.014	94.043					
C16	54.047						
C17	39.024	66.024	74.008				
C174	3.019:	35.035	72.035	94.030			
C177	35.051	63.033	75.032				
C2	8.045	21.009	35.033	36.032	47.050	57.044	76.011
	90.054						
C20	14.026	25.016	25.034	27.038	29.043	35.056	41.016
	47.014	65.015					
C200	13.037	27.007	38.021	41.026	44.051	69.040	72.038
	75.049	77.031	94.031	98.040	104.028		
C212	3.020:	27.008					
C215	21.034	27.012	27.018	29.016	37.038	40.024	44.027
	61.040	61.048	61.061	62.024	67.034	69.011	
C22	3.006:	71.060					
C232	3.021:	44.024					
C240	21.050	36.015	40.027	45.013	56.009	56.044	71.053
	73.029	74.033					
C242	3.022:	37.035	44.011	44.021	73.019	73.033	73.046

C244	38.016	38.026	68.014	68.044	75.028	75.053	
C245	3.023:	28.017	28.041				
C247	3.024:	37.027	73.036	73.038	79.019	80.013	80.033
C250	3.025:	37.007	42.029	44.014	75.016		
C251	3.026:	37.016	42.041	75.020			
C254	3.027:	28.045	29.013	40.018	61.043	61.051	74.031
	76.042	89.018					
C256	3.028:	36.008					
C260	76.014	78.025	78.040				
C262	88.045:	89.023					
C272	3.029:	43.008					
C273	3.030:	23.037					
C275	3.031:	43.029					
C276	3.032:	37.021					
C3	22.011	26.037	57.051	77.040	78.026		
C300	37.030	42.037	51.036	56.041	75.042		
C32	3.007:	22.026	50.033	70.025			
C331	3.033:	39.052	77.036				
C333	3.034:	37.010	45.035	56.038			
C334	44.029	44.048	78.020	78.044			
C335	3.035:	37.013	75.025				
C336	3.036:	21.025	37.024	45.031			
C340	3.038:	41.029	72.041				
C342	3.039:	21.031	37.041	72.047	93.035	94.025	
C343	3.040:	39.034					
C344	3.041:	42.046					

C350	3.042:	73.007					
C351	3.043:	44.008	93.040				
C352	3.044:	43.028	95.022				
C353	3.045:	43.043					
C357	3.046:	43.033	43.047	43.054	95.019		
C361	3.047:	43.018					
C362	3.048:	41.024					
C364	3.049:	43.024					
C365	3.050:	37.033					
C366	3.051:	42.008					
C37	27.039	45.017	75.010				
C370	3.052:						
C374	3.053:	43.060					
C375	3.054:	72.044	79.016	79.035	80.016	94.048	
C376	3.055:	43.013					
C377	15.046 106.047	45.025	51.022	55.022	72.012	73.030	92.036
C4	35.012	44.032	62.017				
C400	71.029	74.047					
C5	12.021	90.031					
C50	99.040	100.045:					
C6	23.053 65.035	28.014 89.008	28.047	35.025	36.022	52.027	56.023
C60	3.008:	27.042	42.056	63.038	71.058	78.012	
C67	63.058	63.062:					
C7	44.040	59.053	65.030				
C71	3.009:	45.029	63.035				

```
EDEXF      39.057   42.015:
EDEXG      27.055   40.007:
EDEXK      35.009:  36.035   36.045
EDEXL      33.047   41.024:
EDEXN      39.058   42.051:
EDEXO      39.007   43.008:  43.015
EDEXQ       4.018   44.008:  44.019   44.022   44.028   44.031   44.033
           44.034   44.038   44.042   44.050   44.054
EDEXR      33.040   42.008:
EDEXS       4.019   35.055   37.034   38.018   38.039:  39.054   45.020
EDEXT      39.059   42.046:  42.057
EDEXV      36.050   38.008:  38.035
EDEXW      38.012   39.011:
EDEXX       4.020   43.060:
EDEXZ      43.059:  44.013
EDITC      27.041   29.023:
EDITI      23.036   27.031   27.034   28.009:  39.048
EDITK      27.042:  39.050
EDITL      21.046:  21.052   27.058
EDITO      23.052:  27.060   39.049
EDITR       4.015   21.033:  21.038   40.030   45.038
EDITS      23.037:  23.049   27.059
EDM8A      52.050   52.053:
EDML        4.014    8.029   47.009:  47.052
EDML1      50.037:  50.051
EDML2      50.039   50.041   50.044   50.047:
```

```
EDML3     51.016:  51.034
EDML4     51.027   51.036:
EDML5     51.010   51.013   51.033   52.006:
EDML6     51.008   51.049   52.014:  52.029
EDML7     52.019   52.031:  52.044
EDML8     52.035   52.046:
EDML9      4.026   52.058   53.006:
EDMLA     48.044:  48.052   49.014   49.015
EDMLB     48.045   49.006:
EDMLD     47.043   47.056   47.059   48.006:
EDMLI     47.038   47.045   47.047:
EDMLN     49.043   50.006:
EDTBL     43.007   57.007:
EDUN       4.016   24.025   24.037   25.022:
EDX3A     37.020   37.027:
EDX3B     37.032   37.035:
EDXEE     40.058   41.009   41.012:
EDXK1     35.011   35.015   35.020   35.024:  36.031
EDXK2     35.022   35.029   35.034:
EDXO1     43.012   43.015:
EDXQ1     44.034:  44.046
EDXQ2     44.037   44.048:
EDXRX     42.012:
EDXW1     35.058   39.017   39.019   39.029   39.033:
EDXW2     39.028   39.052:
EICB      22.018   22.019:  23.011   23.014   23.017   23.024
```

```
P4M         90.010:

PAD.         6.026

PBC.        14.045  14.046  71.031  72.007  72.053  72.056  74.014
            74.028  74.049  76.022  77.046  77.049  93.020  93.049
            93.052  94.008  94.014  94.017  94.021  94.023  94.036
            94.039  95.028  95.029  95.032

PIB          6.024   9.012   9.026  13.052  26.007  67.057  68.037
            72.009  74.016  74.025  76.024  77.051 103.017 105.046
           106.059

PKEY        61.045  61.053  61.061:

PLC.         9.034  10.010  13.009  30.018  53.013  54.012  54.013
            54.022  54.023  54.029  70.061  71.008  71.026  71.032
            71.033  74.040  89.056  93.013  93.021  93.022

PLN70       95.010  95.014:

PLNR        88.049  93.009:

PLNR1       93.012: 93.026  93.039  94.028

PLNR2       93.023: 93.053

PLNR3       93.029  93.055:

PLNR4       93.034: 93.046  94.040  94.055  95.013  95.018  95.021
            95.024  95.033

PLNR5       93.042  94.030:

PLNR6       93.037  93.048: 94.027

PLNR7       94.050  95.007:

PLNR8       95.016  95.019:

PLNR9       94.034  94.042:

PLNRA       94.051: 94.054

PLNRC       94.035: 94.047

PLTBL       84.012: 88.048

PMSK        54.067  54.088:

PNC         88.011  96.019:
```

STYPE	5.023:	26.032	27.061	40.054			
SWPI	6.016:	8.008					
SWPI1	6.024:	7.031					
SWPO	8.009	9.007:	9.030				
SWPO1	9.032	9.036	9.043:				
SWPO2	9.015	9.043	9.048:				
SWTB	4.033	19.013:					
SWTBC	19.042	19.044	19.056:				
SWTBD	19.037	19.051:	19.058				
SZP.	6.025						
TAPE	29.035	70.007:					
TCH	88.039:	89.013	89.017				
TRAPF	7.017	7.026	13.018	15.058	22.034	36.025	68.021
	69.019	80.022					
TRP1	8.016	8.023	8.024:				
TSC.	10.033	66.023					
TSE	5.007:	6.016	6.032	9.007	9.045	17.020	17.025
	17.037	22.038	22.043	28.033	28.053	28.056	29.009
	29.018	35.034	35.040	39.033	39.042	55.006	55.017
	55.029	55.030	55.033	56.007	56.049	68.011	68.039
	68.042	69.039	69.042	74.018	74.019	74.024	74.035
	93.009	93.016					
TSE1	5.008:	15.019	15.036	16.030	16.040	19.016	19.032
	19.045	19.059	20.013	27.023	28.040	28.052	36.026
	36.047	38.008	38.013	38.015	38.024	38.031	38.034
	40.009	40.012	40.013	40.021	47.009	47.041	47.047
	48.011	49.017	49.035	49.041	49.054	50.015	50.042
	51.011	52.006	52.053	56.021	56.035	73.013	73.015
	73.022	73.026	73.043	76.037			
TSE2	5.009:	19.014	19.025	19.027	28.050	29.008	45.008
	45.015	47.023	47.051	50.011	50.034	50.050	56.008
	56.011	56.024	56.025	56.030	56.032	75.039	75.045
	75.052	76.026	76.029	76.032			