

## Appendix G - UT20 Listing

```

!M      0000 ;      0001
0000 ;      0002 ..   UT20 IS A UTILITY PROGRAM USED TO ALTER
0000 ;      0003 ..   MEMORY, DUMP MEMORY, AND BEGIN PROGRAM
0000 ;      0004 ..   EXECUTION AT A GIVEN LOCATION.  THE COMMANDS
0000 ;      0005 ..   ACCEPTED ARE $PHHHH (BEGIN EXECUTION AT THE
0000 ;      0006 ..   SPECIFIED LOCATION WITH RO AS PROGRAM
0000 ;      0007 ..   COUNTER), !MHHHH DATA (PUT DATA AT SPECIFIED
0000 ;      0008 ..   LOCATION), AND ?MHHHH HHHH (OUTPUT DATA
0000 ;      0009 ..   FROM SPECIFIED LOCATION FOR SPECIFIC COUNT).
0000 ;      0010 ..   AT THE BEGINNING OF A COMMAND ALL CHARACTERS
0000 ;      0011 ..   ARE IGNORED UNTIL A ?, !, OR $ IS
0000 ;      0012 ..   ENCOUNTERED.  IN THE ?M AND !M COMMANDS NON
0000 ;      0013 ..   HEX CHARACTERS ARE IGNORED AFTER M UNTIL A
0000 ;      0014 ..   HEX IS READ, THEN THE FIRST NON HEX
0000 ;      0015 ..   CHARACTER MUST BE A SPACE.  NON HEX
0000 ;      0016 ..   CHARACTERS BETWEEN HEX PAIRS OF THE DATA IN
0000 ;      0017 ..   THE !M COMMAND ARE IGNORED EXCEPT FOR CR,
0000 ;      0018 ..   SEMICOLON, AND COMMA.
0000 ;      0019 ..   $L LOADS DATA (WRITTEN IN UT20 FORMAT) FROM
0000 ;      0020 ..   FLOPPY DSK INTO MEMORY.  THERE ARE 77 TRACKS
0000 ;      0021 ..   AVAILABLE ON A DISKETTE (TRACK 0-76).
0000 ;      0022 ..   LOADING STOPS IF THE EOF (DC3) IS DETECTED.
0000 ;      0023 ..   THE BAUD RATE OF UT20 IS DEPENDENT UPON THE
0000 ;      0024 ..   TERMINAL BEING USED.  A CR OR LF IS ENTERED
0000 ;      0025 ..   AT THE BEGINNING TO SPECIFY THE APPROPRIATE
0000 ;      0026 ..   DELAY BETWEEN BITS.  UT20 WILL ECHO
0000 ;      0027 ..   CHARACTERS IF A CR IS CHOSEN AS THE
0000 ;      0028 ..   TIMING CHARACTER.  ECHOING WILL NOT TAKE
0000 ;      0029 ..   PLACE IF A LF IS INPUT AS THE TIMING
0000 ;      0030 ..   CHARACTER.
0000 ;      0031 ..   UT20, AT INITIATION, STORES ALL REGISTERS
0000 ;      0032 ..   BETWEEN WRAM-32 AND WRAM IF IT FINDS RAM
0000 ;      0033 ..   THERE (BUT RO, R1, AND R4.1 ARE CLOBBED).
0000 ;      0034 ..   ?R CAN BE USED TO TYPE THE CONTENTS OF THE 16
0000 ;      0035 ..   REGISTERS (RO-RF).  RO,R1,R4.1 WILL BE
0000 ;      0036 ..   TYPED AS X'S (DON'T CARE).
0000 ;      0037 ..   PTER=#00 ..AUXILIARY FOR MAIN ROUTINE
0000 ;      0038 ..   CL=#01 ..CLOBBED
0000 ;      0039 ..   ST=#02 ..STACK POINTER ONLY REFERENCE TO RAM
0000 ;      0040 ..   SUB=#03 ..SUBROUTINE PROGRAM COUNTER
0000 ;      0041 ..   PC=#05 ..MAIN PROGRAM COUNTER
0000 ;      0042 ..   SWITCH=CL ..DISTINGUISHES BETWEEN ?M AND !M
0000 ;      0043 ..   DELAY=#0C ..DELAY ROUTINE PROGRAM COUNTER
0000 ;      0044 ..   ASL=#0D ..HEX ASSEMBLY REGISTER ON INPUT;
0000 ;      0045 ..   ..AUX FOR HEX OUTPUT
0000 ;      0046 ..   CNTER=ASL ..USED TO COUNT OUTPUT BYTES
0000 ;      0047 ..   AUX=#0E ..AUX.1 HOLDS BIT-TIME CONSTANT
0000 ;      0048 ..   CHAR=#0F ..CHAR.1 HOLDS I/O BYTE
0000 ;      0049 ..   WRAM=#8C1F ..REGISTERS STORED IN RAM
0000 ;      0050 ..   LOADER=#8400 ..LOCATION LOADER PROGRAM
0000 ;      0051 ..
0000 ;      0052 ..   ENTER IN RO
0000 ;      0053 ..
0000 ;      0054 ..   ORG#8000 ..UT20 STARTS AT
8000 ;      0055 ..   ..M(8000)
8000 7100;      0056 ..   DIS,#00 ..P=X=0
8002 F880B0;      0057 ..   LDI A.1(UT20) ;PHI RO ..HOLDS HIGH BIT
8005 ;      0058 ..   ..AFTER FINGER OFF
8005 ;      0059 ..   MAY TRY TO GO TO 8000, NOT 0000
8005 ;      0060 ..   UNTIL FINGER IS OFF BUTTON

```

```

8005 ; 0061 ..
8005 ; 0062 .. THE FOLLOWING WRITES REGISTER CONTENTS INTO
8005 ; 0063 .. WRAM-32 THRU WRAM IF IT EXISTS. WRAM-34 IS
8005 ; 0064 .. ASSUMED NOT TO BE RAM (ELSE ROUTINE OVERRUNS).
8005 ; 0065
8005 ; 0066 ..
8005 F88CB1; 0067 LDI A.1(WRAM) ; PHI CL ..CL IS CLOB-
8008 ; 0068 ..BERED
8008 F81EA1; 0069 LDI A.0(WRAM-1) ; PLO CL ..SET UP WHERE RF.0
800B ; 0070 ..IS TO GO, MINUS 1
800B F8A0B4; 0071 LDI #A0 ; PHI R4 ..R4.1 STORES A
800E ; 0072 ..MODIFIED INSTRUCL.
800E E1; 0073 SEX CL
800F F8D051; 0074 LOOP2: LDI #D0 ; STR CL ..SET UP SEP INSTR.
8012 ; 0075 ..FOR RETURN
8012 F3; 0076 XOR ..CHECK IT WROTE
8013 3A29; 0077 BNZ UT20
8015 21; 0078 DEC CL ..PREPARE FOR MODI-
8016 ; 0079 ..FIED INSTRUCTION
8016 94FC70; 0080 GHI R4 ; ADI #70 ..IN THE 90'S?
8019 331D; 0081 BDF*+ #04
801B FC21; 0082 ADI #21 ..NO, 8N -> 9N
801D FC7F; 0083 ADI #7F ..YES, 9N -> 8(N-1)
801F B451; 0084 PHI R4 ; STR CL ..SET MODIFIED
8021 ; 0085 ..INSTRUC INTO RAM
8021 D1; 0086 SEP CL ..EXECUTE INSTRUCL
8022 ; 0087 ..(80-9F)
8022 73; 0088 STXD ..STORE RESULT RAM
8023 21; 0089 DEC CL ..& BACK UP FOR
8024 94FB90; 0090 GHI R4 ; XRI#90 ..CK IF STORAGE DONE
8027 3A0F; 0091 BNZ LOOP2 ..NEXT BYTE
8029 ; 0092 ..
8029 90B5B3; 0093 UT20: GHI R0 ; PHI PC ; PHI SUB ..#80->PC.1 & SUB.1
802C F830A5; 0094 LDI A.0(UT20A) ; PLO PC
802F D5; 0095 SEP PC
8030 E5; 0096 UT20A: SEX PC
8031 7155; 0097 DIS #55 ..NOTE PC=5 ASSUMED
8033 6101; 0098 OUT 1, #01 ..SELECT RCA GROUP
8035 F88CB2; 0099 LDI A.1(WRAM) ; PHI ST ..SET STACK POINTER
8038 F800A2; 0100 LDI #00 ; PLO ST
803B ; 0101 ..TO M(8C00), ONLY
803B ; 0102 ..RAM USED
803B F8FEA3; 0103 LDI A.0(TIMALC) ; PLO SUB ..READ ONE CHAR
803E ; 0104 ..TO SET TIMER
803E D3; 0105 SEP SUB
803F ; 0106 ..
803F ; 0107 ... INITIATION NOW DONE
803F ; 0108 ..
803F F89CA3; 0109 START: LDI A.0(TYPE5D) ; PLO SUB
8042 F881B3; 0110 LDI A.1(TYPE5D) ; PHI SUB
8045 D30D; 0111 SEP SUB ; #0D ..CR=CARRIAGE RET
8047 D30A; 0112 ST2: SEP SUB ; #0A ..LF=LINE FEED
8049 D32A; 0113 SEP SUB ; #2A ..* PROMPT CHARAC
804B F800ADB0; 0114 IGNORE: LDI #00 ; PLO ASL ; PHI ASL ..PREPARE TO INPUT
804F ; 0115 ..HEX DIGITS,
804F ; 0116 ..CLEAR ASL
804F F83BA3; 0117 LDI A.0(READAH) ; PLO SUB
8052 D3; 0118 SEP SUB ..INPUT COMMAND
8053 FB24; 0119 XRI #24 ..IS IT "$" ?
8055 C28207; 0120 LBZ DOLLAR
8058 FB05; 0121 XRI #05 ..IS IT "!" ?
805A ; 0122 ..TEST $ XRI !

```

```

805A A1;          0123      PLO SWITCH          ..AND SAVE RESULT
805B CE;          0124      LSZ
805C FB1E;        0125      XRI #1E              ..IS IT "?" ?
805E ;            0126
805E 3A4B;        0127      BNZ IGNORE          ..TES $ XRI ! XRI ?
8060 ;            0128
8060 ;            0129      ..
8060 ;            0130      .. THE FOLLOWING IS COMMON FOR ?M AND !M
8060 ;            0131      .. (SWITCH.0 = 0 FOR THE LATTER)
8060 ;            0132      ..
8060 D3;          0133      RDARGS: SEP SUB          ..NOTE SUB AT
8061 ;            0134
8061 ;            0135
8061 FB4D;        0136      XRI #4D              ..READAH. READ
8063 3ADC;        0137      BNZ ISITR            ..HEX ARGUMENTS
8065 D3;          0138      RD1: SEP SUB          ..SHOULD BE "M"
8066 3B65;        0139      BNF *-#01            ..CK FOR ?R
8068 ;            0140
8068 D3;          0141      SEP SUB              ..IGNORE NON HEX
8069 3368;        0142      BDF *-#01            ..CHARS. AFTER "M"
806B ;            0143
806B 9DB0;        0144      GHI ASL ; PHI PTER      ..READ FIRST ARG
806D 8DA0;        0145      GLO ASL ; PLO PTER      ..(LOCA. IN MEMORY)
806F ;            0146
806F F800ADB0;    0147      LDI#00 ; PLO ASL ; PHI ASL ..PTER NOW POINTS
8073 1D;          0148      INC ASL              ..TO USER MEMORY
8074 ;            0149
8074 9FFB0D;      0150      GHI RF ; XRI#0D        ..CLEAR ASL
8077 3A7E;        0151      BNZ TEST              ..?MXXXXCR PRINTS
8079 81;          0152      GLO SWITCH            ..TWO HEX DIGITS
807A 3A8D;        0153      BNZ LINE-#03          ..CK FOR CR
807C 30E1;        0154      PR SYNERR            ..BR IF NOT A CR
807E FB2D;        0155      TEST: XRI#2D          ..BR IF ?
8080 3AE1;        0156      BNZ SYNERR            ..OTHERWISE ERROR
8082 2D;          0157      DEC ASL              ..CK FOR SPACE
8083 81;          0158      GLO SWITCH
8084 32C6;        0159      BZ EX1              ..ADJUST ASL
8086 ;            0160
8086 ;            0161      ..
8086 ;            0162      .. THE FOLLOWING DOES (?M LOC COUNT) AND
8086 ;            0163      .. (?MXXXXCR) COMMANDS
8086 D3;          0164      RD2: SEP SUB          ..READ SECOND ARG
8087 3386;        0165      BDF RD2              ..(NUMBER OF BYTES)
8089 ;            0166
8089 FB0D;        0167      XRI #0D              ..NEXT CK FOR CR
808B 3AE1;        0168      BNZ SYNERR
808D F89CA3;      0169      LDI A.0(TYPE5D) ; PLO SUB ..TYPE
8090 D30A;        0170      LINE: SEP SUB; ,#0A    ..LF
8092 90BF;        0171      GHI PTER ; PHI CHAR    ..PREPARE LINE
8094 ;            0172
8094 F8AEA3;      0173      LDI A.0(TYPE2) ; PLO SUB    ..HEADING
8097 D3;          0174      SEP SUB              ..TYPE 2 HEX DIGITS
8098 80BF;        0175      GLO PTER ; PHI CHAR
809A F8AEA3;      0176      LDI A.0(TYPE2) ; PLO SUB
809D D3;          0177      SEP SUB              ..TYPE OTHER TWO
809E D320;        0178      TSPACE: SEP SUB; ,#20    ..SPACE
80A0 ;            0179      ..
80A0 40BF;        0180      TLOOP: LDA PTER ; PHI CHAR ..FETCH ONE BYTE
80A2 ;            0181
80A2 F8AEA3;      0182      LDI A.0(TYPE2) ; PLO SUB    ..FOR TYPING
80A5 D3;          0183      SEP SUB              ..TYPE 2 HEX
80A6 2D;          0184      DEC COUNTER

```

```

80A7 8D;          0185      GLO CINTER
80A8 3AAD;        0186      BNZ TL3          ..BRANCH NOT DONE
80AA 9D;          0187      GHI CINTER
80AB 323F;        0188      BZ START          ..BRANCH IF DONE
80AD 80FA0F;      0189 TL3:  GLO PTER ;ANI #0F    ..PTER DIV BY 16?
80B0 3AB8;        0190      BNZ TL2
80B2 D33B;        0191      SEP SUB; ,#3B        ..YES TYPE ";"
80B4 D30D;        0192      SEP SUB; ,#0D        ..THEN CR
80B6 3090;        0193      BR LINE
80B8 F6;          0194 TL2:  SHR              ..DIV BY 2?
80B9 33A0;        0195      BDF TLOOP          ..NO, LOOP BACK
80BB 309E;        0196      BR TSPACE          ..ELSE TYPE SPACE &
80BD ;           0197      ..LOOP BACK
80BD ;           0198      ..
80BD ;           0199      .. THE FOLLOWING DOES (!M LOC DATA) COMMAND
80BD ;           0200      .. ENTER AT EX1
80BD ;           0201      ..
80BD ;           0202      .. EFFECT OF THE FOLLOWING IS TO READ IN HEX
80BD ;           0203      .. TERMINATING WITH A CR, IGNORING NON-HEX CHAR
80BD ;           0204      .. PAIRS; EXCEPTIONS: A COMMA BEFORE A CR ALLOWS
80BD ;           0205      .. THE INPUT TO CONTINUE ON THE NEXT LINE AND A
80BD ;           0206      .. SEMICOLON ALLOWS THE !M COMMAND TO BE ASSUMED.
80BD ;           0207      ..
80BD D3;          0208 EX3:  SEP SUB              ..INPUT UNTIL A
80BE ;           0209      ..HEX IS READ
80BE 38BD;        0210      BNF EX3
80C0 D3;          0211 EX2:  SEP SUB              ..LOOK FOR SECOND
80C1 ;           0212      ..HEX DIGIT
80C1 3BE1;        0213      BNF SYNERR          ..BR IF NOT HEX
80C3 8D50;        0214      GLO ASL ;STR PTER    ...*SET BYTE*
80C5 10;          0215      INC PTER
80C6 D3;          0216 EX1:  SEP SUB              ..NOTE SUB @ READAH
80C7 33C0;        0217      BDF EX2              ..BRANCH IF HEX
80C9 FB0D;        0218      XRI #0D              ..CHECK IF CR
80CB 323F;        0219      BZ START
80CD FB21;        0220 EX4:  XRI #21              ..ELSE CK FOR COMMA
80CF ;           0221      ..(TEST CR XRI ",")
80CF 32BD;        0222      BZ EX3              ..IF ELSE BRANCH
80D1 FB17;        0223      XRI #17              ..ELSE CK FOR ";"
80D3 ;           0224      ..(TEST CR XRI
80D3 ;           0225      ..", " XRI ";")
80D3 3AC6;        0226      BNZ EX1              ..IGNORE ALL ELSE
80D5 D3;          0227      SEP SUB              ..ON ";" IGNORE ALL
80D6 ;           0228      ..UNTIL CR, THEN
80D6 ;           0229      ..LOOP BACK
80D6 FB0D;        0230      XRI #0D
80D8 3AD5;        0231      BNZ *-03
80DA 3065;        0232      BR RD1              ..THEN BRANCH BACK
80DC ;           0233      ..FOR !M COMMAND
80DC FB1F;        0234 ISITR: XRI#1F              ..IS IT R?
80DE C282E8;      0235      LBZ TYPER          ..BR IF R
80E1 ;           0236      ..
80E1 F89CA3;      0237 SYNERR: LDI A.0(TYPE5D);PLO SUB ..GENERAL RESULT
80E4 ;           0238      ..SYNTACTIC ERROR
80E4 D30D;        0239      SEP SUB; ,#0D        ..CR
80E6 C08200;      0240      LBR FSYNER
80E9 ;           0241      ..
80E9 ;           0242      ..
80E9 ;           0243      ..
80E9 ;           0244      .. SUBROUTINES
80E9 ;           0245      ..
80E9 ;           0246      ..

```

```

80E9 ;
80EA ;
80EA ;
80EA ;
80EA ;
80EA ;
80EA ;
80EA ;
80EA ;
80EA ;
80EA DCD CDCDC;
80EE D3;
80EF 9EF6AE;
80F2 ;
80F2 2E;
80F3 ;
80F3 43FF01;
80F6 3AF4;
80F8 ;
80F8 8E;
80F9 32EA;
80FB 23;
80FC ;
80FC 30F2;
80FE ;
80FE ;
80FE ;
80FE ;
80FE ;
80FE ;
80FE ;
80FE ;
80FE ;
80FE ;
80FE 93BC;
8100 F8EFAC;
8103 F800AEAF;
8107 ;
8107 3707;
8109 3F09;
810B ;
810B F803;
810D ;
810D ;
810D FF01;
810F 3A0D;
8111 8F;
8112 ;
8112 ;
8112 3A17;
8114 3719;
8116 ;
8116 ;
8116 1F;
8117 371E;
8119 ;
8119 1E;
811A F807;
811C ;
811C 300D;
811E ;
811E ;
811E ;

0247      ORG*+ #01
0248 ..   DELAY ROUTINE
0249 ..   DELAY IS 2(1+AUX.1(3+@SUB))
0250 ..   USED BY TYPE, READ, AND TIMALC.
0251 ..   AUX.1 IS ASSUMED TO HOLD A DELAY CONSTANT
0252 ..   =((BIT TIME OF TERMINAL)/
0253 ..   (20*INSTR TIME OF COSMAC))-1.
0254 ..   THIS CONSTANT CAN BE GENERATED
0255 ..   AUTOMATICALLY BY THE TIMALC ROUTINE.
0256 ..
0257 DEXIT: SEP RC;SEP RC;SEP RC;SEP RC      ..4 NOP'S
0258      SEP SUB      ..RETURN
0259 DELAY1: GHI AUX ;SHR ;PLO AUX      ..SHIFT OUT
0260      ..ECHO FLAG
0261 DELAY2: DEC AUX      ..AUX.0 HOLDS BASIC
0262      ..BIT DELAY
0263      LDA SUB ;SMI #01      ..PICK UP CONSTANT
0264      BNZ *-#02      ..LOOP AS SPECIFIED
0265      ..BY CALL
0266      GLO AUX      ..DONE YET?
0267      BZ DEXIT
0268      DEC SUB      ..POINTS SUB AT
0269      ..DELAY POINTER
0270      BR DELAY2
0271 ..
0272 ..   ROUTINE TO CALCULATE BYTE TIME AND ECHO
0273 ..   FLAG.  WAITS FOR LF(NO ECHO) OR CR(ECHO)
0274 ..   TO BE TYPED IN.  ALSO SETS UP POINTER TO
0275 ..   DELAY ROUTINE.
0276 ..   AUX.1 ENDS UP HOLDING, IN THE MOST
0277 ..   SIGNIFICANT 7 BITS, THE DELAY CONSTANT.
0278 ..   LEAST SIGNIFICANT BIT IS ZERO FOR ECHO,
0279 ..   ONE FOR NO ECHO.
0280 ..
0281 TIMALC: GHI SUB ;PHI DELAY
0282      LDI A.0(DELAY1) ;PLO DELAY
0283      LDI #00 ;PLO AUX ;PLO CHAR
0284      ..DELAY ROUT. READY
0285      B4*      ..WAIT START BIT
0286      BN4*      ..WAIT FOR FIRST
0287      ..NONZERO DATA BIT
0288      LDI #03      ..SET UP FOR 10
0289      ..EXECUTIONS SO
0290      ..ROUND-OFF MINIMAL
0291 TC2: SMI #01
0292      BNZ *-#02
0293      GLO CHAR      ..LOOK TO SEE IF
0294      ..DATA CHANGED
0295      ..PREVIOUSLY
0296      BNZ ZRONE      ..BR IF IT 6HAD
0297      B4 INCR      ..ELSE LOOK FOR
0298      ..CHANGE TO ZERO
0299      ..BRANCH IF NO
0300      INC CHAR      ..YES, SET SWITCH
0300 ZRONE: B4 DAUX      ..LOOK FOR CHANGE
0301      ..TO 1, BR IF YES
0302 INCR: INC AUX
0303      LDI #07      ..SET UP FOR 20
0304      ..INSTRUCTION LOOPS
0305      BR TC2
0306 ..
0307 ..   AUX.0 NOW HOLDS #LOOPS IN 2 BIT TIMES
0308 ..

```



```

811E 2E2E;          0309 DAUX:  DEC AUX ;DEC AUX          ..REDUCE COUNT TO
8120 ;              0310                      ..BALANCE FIXED
8120 ;              0311                      ..OVERLOAD IN
8120 ;              0312                      ..CALLING DELAY
8120 8EF901BE;      0313          GLO AUX ;ORI #01 ;PHI AUX..LSB AUX.1 = 1.5
8124 DC0C;          0314          SEP RC; ,#0C          ..BIT TIME DELAY
8126 3F2C;          0315          BN4 WAIT          ..ER IF LF(NO ECHO)
8128 ;              0316                      ..LSB AUX.1=1
8128 9EFAFE;        0317          GHI AUX ;ANI#FE
812E BE;            0318          PHI AUX          ..CR(ECHO)
812C ;              0319                      ..LSB AUX.1=0
812C DC26;          0320 WAIT:  SEP RC; ,#26
812E D5;            0321          SEP R5
812F ;              0322 ..
812F ;              0323 ..
812F ;              0324 .. READ ROUTINE--READS 1 BYTE INTO CHAR.1. WHEN
812F ;              0325 .. ENTERED VIA READAH, THEN IF INPUT IS A HEX
812F ;              0326 .. DIGIT ITS HEX VALUE IS SHIFTED INTO ASL FROM
812F ;              0327 .. THE RIGHT AND DF=1, ELSE DF=0; CLOBBERS CHAR,
812F ;              0328 .. AUX.0, (ASL ON READAH). LEAVES BYTE IN D
812F ;              0329 .. (BUT CLOBBED IF SUBR LINKAGE IS USED).
812F ;              0330 .. LEAVES PC AT READAH ENTRY POINT; EXITS TO R5.
812F ;              0331 ..
812F ;              0332 ..
812F ;              0333 .. WARNING: READ PROCESS HAS NOT FINISHED. DO
812F ;              0334 .. NOT TYPE IMMEDIATELY, OR ELSE ENTER TYPE VIA
812F ;              0335 .. TYPESD.
812F ;              0336 ..
812F ;              0337 ..
812F FC07;          0338 CKDEC:  ADI #07          ..CK FOR ASCII
8131 ;              0339                      ..DECIMAL INPUT
8131 3337;          0340          BDF NFND
8133 FCOA;          0341          ADI #0A
8135 3377;          0342          BDF FND          ..SUB NET 30
8137 FC00;          0343 NFND:  ADI #00          ..SETS DF=0
8139 9F;            0344 REXIT:  GHI CHAR          ..CHARACTER INTO D
813A D5;            0345          SEP R5
813B F800;          0346 READAH: LDI #00
813D 38;            0347          SKP          ..SKIP OVER
813E ;              0348                      ..TO READ1
813E 93;            0349 READ:  GHI SUB          ..CONSTANT WITH
813F ;              0350                      ..A VALUE > 0
813F AF;            0351 READ1:  PLO CHAR          ..SET ENTRY FLAG
8140 F880BF;        0352 READ2:  LDI #80 ;PHI CHAR          ..INITIALIZE INPUT
8143 ;              0353                      ..BYTE-WHEN SHIFTED
8143 ;              0354                      ..80 IS 1, WILL BE
8143 ;              0355                      ..DONE
8143 E2;            0356          SEX ST
8144 3F44;          0357          BN4 *          ..WAIT FOR END OF
8146 ;              0358                      ..LAST DATA BIT
8146 3746;          0359          B4 *          ..WAIT FOR PRESENT
8148 ;              0360                      ..START BIT
8148 DC02;          0361          SEP RC; ,#02          ..DELAY HALF
814A ;              0362                      ..BIT TIME
814A ;              0363 ..
814A F80052;        0364 NOBIT:  LDI #00 ;STR ST
814D 9EFA01;        0365 LOOP5:  GHI AUX ;ANI #01          ..CHECK IF ECHO
8150 ;              0366                      ..INDICATOR IS
8150 ;              0367                      ..LSB OF AUX.1
8150 F152;          0368          OR ;STR ST          ..OUTPUT IS ONE(NO
8152 ;              0369                      ..EFFECT) ON NOECHO
8152 6722;          0370          OUT 7 ;DEC ST

```

```

8154 ; 0371 ..
8154 DC07; 0372 LOOP5B: SEP RC; ,#07 ..DELAY ONE
8156 ; 0373 ..BIT TIME
8156 F80152; 0374 LDI #01 ;STR ST
8159 9FF6BF; 0375 GHI CHAR ;SHR ;PHI CHAR ..SHIFT INPUT CHAR.
815C 3365; 0376 BDF NEXT ..BR IF INPUT
815E ; 0377 ..FINISHED D=CHAR.1
815E F980; 0378 ORI #80
8160 3F4A; 0379 BN4 NOBIT ..BR IF INPUT
8162 ; 0380 ..BIT A ZERO
8162 BF; 0381 PHI CHAR ..ELSE PUT OK'D
8163 ; 0382 ..VALUE AWAY
8163 304D; 0383 BR LOOP5
8165 ; 0384 ..
8165 ; 0385 .. NOW HAVE BYTE READ INTO CHAR.1
8165 ; 0386 ..
8165 6722; 0387 NEXT: OUT 7; DEC ST ..OUTPUT STOP BIT
8167 324D; 0388 BZ READ2 ..BR IF D=0,
8169 ; 0389 ..CHAR.1 IS A NULL
8169 8F; 0390 GLO CHAR ..CHECK ENTRY FLAG
816A 3A39; 0391 BNZ REXIT ..BR IF ENTRY VIA
816C ; 0392 ..READ
816C 9F; 0393 CKHXE: GHI CHAR
816D FF41; 0394 SMI #41 ..CK FOR ASCII HEX
816F 3B2F; 0395 BNF CKDEC ..AT TOP OF ROUTINE
8171 FFD6; 0396 SMI #06 ..CK FOR A THRU F
8173 3337; 0397 BDF NFND
8175 FC10; 0398 ADI #10 ..SUB NET 37
8177 ; 0399 ..
8177 AE; 0400 FND: PLO AUX ..SAVE TO SHIFT
8178 ; 0401 ..INTO ASL
8178 9D; 0402 GHI ASL
8179 FEFEFefe; 0403 SHL ;SHL ;SHL ;SHL ..SHIFT ASL.1
817D ; 0404 ..LEFT FOUR
817D 52; 0405 STR ST
817E 8D; 0406 GLO ASL
817F F6F6F6F6; 0407 SHR ;SHR ;SHR ;SHR ..SHIFT ASL.0 RT 4
8183 F1BD; 0408 OR ;PHI ASL ..COMBINE
8185 8D; 0409 GLO ASL
8186 FEFEFefe; 0410 SHL ;SHL ;SHL ;SHL ..SHIFT ASL.0
818A ; 0411 ..LEFT FOUR
818A 52; 0412 STR ST
818B 8EFAOFF1AD; 0413 GLO AUX ;ANI #0F ;OR ;PLO ASL ..COMBINE
8190 FF00; 0414 SMI #00 ..SET DF
8192 3039; 0415 BR REXIT
8194 ; 0416 ..
8194 ; 0417 .. TYPE ROUTINE -- TYPES 1 BYTE FROM @R5!, @R6!,
8194 ; 0418 .. OR CHAR.1, OR TYPES A BYTE AS 2 HEX DIGITS FROM
8194 ; 0419 .. CHAR.1 FOLLOWS A LINE FEED BY SIX NULLS.
8194 ; 0420 .. USES 2 AUXILIARY REGS - AUX AND CHAR - PLUS
8194 ; 0421 .. RAM LOCATION @ST. EXITS READY TO TYPE 1 BYTE
8194 ; 0422 .. FROM @R5!. EXITS TO R5 WHEN ENTERED AT TYPE5D,
8194 ; 0423 .. PAUSES TO ALLOW AN EARLIER READ TO COMPLETE.
8194 ; 0424 ..
8194 ; 0425 .. AUX.0 HOLDS OUTPUT CHAR (AT FIRST), THEN THE
8194 ; 0426 .. DELAY CONSTANT BETWEEN BITS. CHAR.0 HOLDS THE
8194 ; 0427 .. NUMBER OF BITS (11) IN ITS LOWER DIGIT, AND
8194 ; 0428 .. IN ITS UPPER DIGIT HOLDS A CODE --
8194 ; 0429 .. 0 FOR BYTE OUTPUT
8194 ; 0430 .. 1 FOR FIRST HEX OUTPUT
8194 ; 0431 .. 2 FOR LAST NULL OUTPUT
8194 ; 0432 .. 8 FOR LF OUTPUT

```

8194 ;	0433 ..	
8194 ;	0434 ..	
8194 ;	0435	ORG #819C
819C DC17;	0436 TYPE5D: SEP RC; ,#17	..3 BIT TIME DELAY
819E 38;	0437 SKP	..SKIP TO TYPE5
819F D5;	0438 TEXTIT: SEP R5	
81A0 4538;	0439 TYPE5: LDA R5 ;SKP	..ENTRY FOR UT20
81A2 ;	0440	..SKIP TO TYPE
81A2 4638;	0441 TYPE6: LDA R6 ;SKP	..ENTRY FOR G.P.
81A4 ;	0442	..IMMED,TH
81A4 9F;	0443 TYPE: GHI CHAR	
81A5 AE;	0444 TY1: PLO AUX	..SAVE BYTE
81A6 FB0A;	0445 XRI#0A	..IS IT LINE FEED?
81A8 3AC0;	0446 BNZ TY2	
81AA F88B;	0447 LDI#8B	..(# BITS)+(# NULLS
81AC ;	0448	..TO FOLLOW LF + 1)
81AC 30C2;	0449 BR TY3	
81AE 9F;	0450 TYPE2: GHI CHAR	..UT20 ENTRY
81AF F6F6F6F6;	0451 TY4: SHR ;SHR ;SHR ;SHR	..SHIFT FIRST HEX
81B3 ;	0452	..TO THE RIGHT
81B3 FCF6;	0453 ADI#F6	..CONVERT TO HEX
81B5 3BB9;	0454 BNF *+ #04	..IF "A" OR MORE
81B7 FC07;	0455 ADI#07	..ADD NET 37
81B9 FFC6AE;	0456 SMI#C6 ;PLO AUX	..ELSE ADD NET 30
81BC F81B;	0457 LDI#1B	..10+(# OF BITS)
81BE 30C2;	0458 BR TY3	
81C0 ;	0459 ..	
81C0 F80B;	0460 TY2: LDI#0B	..#BITS TO OUTPUT
81C2 AF;	0461 TY3: PLO CHAR	..SAVE MAIN TALLY
81C3 ;	0462	..VALUE
81C3 E2;	0463 SEX ST	
81C4 ;	0464 ..	
81C4 F80052;	0465 BEGIN: LDI#00 ;STR ST	..FOR START BIT
81C7 67;	0466 OUT 7	
81C8 22;	0467 DEC ST	..BACK TO WHERE
81C9 ;	0468	..IT WAS
81C9 8E;	0469 GLO AUX	..PUT CHAR BACK
81CA 52;	0470 PREBIT: STR ST	
81CB DC07;	0471 BITS: SEP RC; ,#07	..DELAY 1 BIT TIME
81CD 2F;	0472 DEC CHAR	..DECREMENT TALLY
81CE FOAEFA0152;	0473 LDX ;PLO AUX ;ANI#01 ;STR ST	
81D3 67;	0474 OUT 7	..OUTPUT DATA BIT
81D4 22;	0475 DEC ST	
81D5 8FFA0F;	0476 GLO CHAR ;ANI#0F	
81D8 32E28E;	0477 BZ NXCHAR ;GLO AUX	..AUX.0 TO STRETCH
81DB ;	0478	..DELAY
81DB 8EF6F980;	0479 GLO AUX ;SHR ;ORI#80	..SHIFT TO
81DF 52;	0480 STR ST	..NEXT BIT
81E0 30CA;	0481 BR PREBIT	
81E2 ;	0482 ..	
81E2 8FFCFB;	0483 NXCHAR: GLO CHAR ;ADI#FB	..SET UP FOR
81E5 AF;	0484 PLO CHAR	..NEXT CHAR
81E6 3B9F;	0485 BNF TEXTIT	..EXIT IF NO MORE
81E8 FF1B;	0486 SMI#1B	..TEST FOR
81EA ;	0487	..ALTERNATIVES
81EA 329F;	0488 BZ TEXTIT	..IF JUST TYPED
81EC ;	0489	..LAST NULL
81EC 3BF2;	0490 BNF HEX2	..IF JUST TYPED
81EE ;	0491	..FIRST NULL, LF
81EE ;	0492	..OR NULL
81EE F800;	0493 LDI#00	..PREPARE TO TYPE
81FO ;	0494	..NULL



```

81F0 30FD;      0495          BR HX22
81F2 ;          0496          ..
81F2 9FFA0F;    0497 HEX2:   GHI CHAR ;ANI#OF      ..GET SECOND HEX
81F5 ;          0498          ..DIGIT
81F5 FCF6;      0499          ADI#F6              ..CONVERT TO HEX
81F7 3BFB;      0500          BNF *+#04           ..IF "A" OR MORE
81F9 FC07;      0501          ADI#07              ..ADD NET 37
81FB FFC6;      0502          SMI#C6              ..ELSE ALL NET 30
81FD AE;        0503 HX22:   PLO AUX              ..STORE CHAR AWAY
81FE 30C4;      0504          BR BEGIN
8200 ;          0505          ..
8200 ;          0506          ..
8200 D30A;      0507 FSYNER: SEP SUB; ,#0A          ..LF
8202 D33F;      0508          SEP SUB; ,#3F          ...?
8204 C0803F;    0509          LBR START
8207 ;          0510          ..
8207 ;          0511          .. THE FOLLOWING DOES $P HHHH , $U HHHH
8207 ;          0512          ..
8207 F85FA1;    0513 DOLLAR:LDI A.0(INTRPT) ;PLO R1 ..R1 IS POINTING
820A F882B1;    0514          LDI A.1(INTRPT) ;PHI R1 ..AT INTRPT
820D D3;        0515          SEP SUB              ..SUB.0=READAH
820E FB55;      0516          XRI #55              ..CHECK FOR "U"
8210 3231;      0517          BZ D1              ..CON'T WITH "U"
8212 FB19;      0518          XRI #19              ..CHECK FOR "L"
8214 3245;      0519          BZ DOLL              ..IF "L"
8216 FB1C;      0520          XRI#1C              ..CK FOR "P"
8218 CA80E1;    0521          LBNZ SYNERR          ..NOT P EITHER
821E D3;        0522          SEP SUB
821C 331B;      0523          BDF *-#01              ..ASSEMBLE HEX
821E ;          0524          ..STRING INTO ASL
821E FB0D;      0525          XRI #0D              ..FIRST NON-HEX
8220 ;          0526          ..MUST BE CR
8220 CA80E1;    0527          LBNZ SYNERR
8223 F89CA3;    0528          LDI A.0(TYPE5D) ;PLO SUB
8226 D30A;      0529          SEP SUB; ,#0A          ..LF
8228 E5;        0530          SEX PC
8229 7055;      0531          RET,#55
822B 6100;      0532          OUT 1,#00              ..CLEAR I/O DECODER
822D 6704;      0533          OUT 7,#04              ..BIT 2 DESELECT!
822F ;          0534          ..THE 2 LEVEL I/O
822F 3039;      0535          BR D2
8231 D3;        0536 D1:   SEP SUB
8232 3331;      0537          BDF D1              ..ASSEMBLE HEX
8234 ;          0538          ..STRING INTO ASL
8234 FB0D;      0539          XRI #0D              ..FIRST NON-HEX
8236 ;          0540          ..MUST BE CR
8236 CA80E1;    0541          LBNZ SYNERR
8239 9DB0;      0542 D2:   GHI ASL ;PHI RO
823B 8DA0;      0543          GLO ASL ;PLO RO          ..SET UP NEXT PC
823D F89CA3;    0544          LDI A.0(TYPE5D) ;PLO SUB
8240 D30A;      0545          SEP SUB; ,#0A          ..LF
8242 E5;        0546          SEX PC
8243 7000;      0547          RET, #00              ..AND USER PROGRAM
8245 ;          0548          ..BEGINS (IN RO)
8245 ;          0549          ..EXIT TO UT20
8245 ;          0550          .. THE FOLLOWING DOES $L
8245 F800A0;    0551 DOLL:LDI A.0(LOADER) ;PLO RO
8248 F884B0;    0552          LDI A.1(LOADER) ;PHI RO
824B E0;        0553          SEX RO
824C D0;        0554          SEP RO
824D ;          0555          .. MSGE ROUTINE
824D ;          0556          .. THIS ROUTINE INITIALIZES RC TO
824D ;          0557          .. POINT AT THE DELAY ROUTINE.

```

```

824D ;
824D ;
824D F8EFAC;
8250 F880BC;
8253 DC12;
8255 46BF;
8257 325E;
8259 D481A4;
825C 3055;
825E D5;
825F ;
825F ;
825F ;
825F ;
825F ;
825F F864A4;
8262 ;
8262 F874A5;
8265 ;
8265 F883B4;
8268 B5;
8269 F81FA2;
826C F88CB2;
826F F876A3;
8272 F882B3;
8275 D3;
8276 D4824D;
8279 494E5452505421;
8280 00;
8281 ;
8281 ;
8281 ;
8281 ;
8281 ;
8281 ;
8281 E3;
8282 F8EFAC;
8285 ;
8285 ;
8285 F880BC;
8288 F83FA5;
828B F880B5;
828E F800A2;
8291 F88CB2;
8294 6101;
8296 ;
8296 7155;
8298 ;
8298 ;
8298 ;
8298 ;
8298 ;
8298 ;
8298 F805A3;
829B F800B3;
829E ;
829E F864A4;
82A1 ;
82A1 ;

0558 .. IT TYPES OUT DATA POINTED TO BY R6. THIS
0559 .. ROUTINE USES THE STANDARD CALL AND RET ROUTINES.
0560 MSGE:LDI A.0(Delay1) ;PLO RC
0561 LDI A.1(Delay1) ;PHI RC
0562 SEP RC,#12 ..DELAY
0563 STRNG:LDI A.0(RF) ;PHI RF ..LOAD CHAR TO RF.1
0564 BZ EXIT1
0565 SEP R4; ,A(TYPE) ..TYPE OUT CHAR
0566 BR STRNG
0567 EXIT1:SEP R5
0568 .. INTERRUPT ROUTINE
0569 .. IT INITIALIZES R4,R5 TO POINT AT
0570 .. THE CALL AND RETURN ROUTINES. IT CALLS OSTRNG,
0571 .. AND OUTPUT 'INTRPT ON' MESSAGE.
0572 .. IT EXITS OSTRNG WITH R3 AS PROGRAM COUNTER,
0573 .. THEN IT TRANSFERS CONTROL TO UT20.
0574 INTRPT:LDI A.0(CALL) ;PLO R4 ..INITIALIZE CALL
0575 ..POINTER
0576 LDI A.0(RET) ;PLO R5 ..INITIALIZE RET
0577 ..POINTER
0578 LDI A.1(CALL) ;PHI R4 ..CALL AND RET ON
0579 PHI R5 ..SAME PAGE
0580 LDI#1F ;PLO R2 ..INITIALIZE I/O
0581 LDI A.1(WRAM) ;PHI R2 ..POINTER
0582 LDI A.0(MSG) ;PLO R3 ..INITIALIZE PC
0583 LDI A.1(MSG) ;PHI R3
0584 SEP R3
0585 MSG:SEP R4; ,A(MSGE)
0586 ,T'INTRPT!','#00
0587 .. ENTER ROUTINE
0588 .. THIS ROUTINE INITIALIZES RC TO POINT AT
0589 .. THE DELAY ROUTINE. IT ALSO INITIALIZES R2 TO
0590 .. LOC #8C00 (THE I/O LOCATION USED BY UT20).
0591 .. IT DISABLES INTERRUPT, SELECTS RCA I/O
0592 .. GROUP, AND TRANSFERS CONTROL TO UT20.
0593 ENTER:SEX R3 ..X=P=3
0594 LDI A.0(Delay1) ;PLO RC ..INITIALIZE RC
0595 ..TO POINT AT THE
0596 ..DELAY ROUTINE
0597 LDI A.1(Delay1) ;PHI RC
0598 LDI A.0(START) ;PLO R5 ..INITIALIZE PC
0599 LDI A.1(START) ;PHI R5
0600 LDI#00 ;PLO R2 ..R2 POINTS TO
0601 LDI A.1(WRAM) ;PHI R2 ..M(8C00)
0602 OUT 1,#01 ..SELECT RCA I/O
0603 ..GROUP
0604 DIS,#55 ..DISABLE INTRPT
0605 ..P=X=5
0606 .. DSKGO ROUTINE
0607 .. THIS ROUTINE INITIALIZES R4,R5,RC TO
0608 .. POINT AT THE CALL, RET, AND DELAY
0609 .. ROUTINES RESPECTIVELY.
0610 .. THIS ROUTINE DYNAMICALLY DETERMINES THE
0611 .. STACK LOCATION,AND INITIALIZE R2 TO
0612 .. POINT AT THAT LOCATION.
0613 .. IT ALSO HOMES BOTH DSK DRIVES IF POSSIBLE.
0614 DSKG01:LDI#05 ;PLO R3
0615 LDI#00 ;PHI R3
0616 DSKG02:ORG *
0617 LDI A.0(CALL) ;PLO R4 ..INITIALIZE R4 TO
0618 ..POINT AT THE
0619 ..CALL ROUTINE

```

```

82A1 F874A5;      0620          LDI A.0(RET)      ;PLO R5      ..INITIALIZE R5 TO
82A4 ;            0621                      ..POINT AT RET ROU-
82A4 ;            0622                      ..TINE
82A4 F883B4;      0623          LDI A.1(CALL)    ;PHI R4      ..R4,R5 ON
82A7 B5;          0624          PHI R5          ..SAME PAGE
82A8 F880B2;      0625          LDI#80          ;PHI R2
82AB F8FFA2;      0626          LDI#FF          ;PLO R2
82AE 92FF01B2;    0627  STACK:GHI R2      ;SMI#01      ;PHI R2  ..ASSUME 4K BANKS
82B2 ;            0628                      ..OF MEMORY
82B2 F85A5202;    0629          LDI#5A          ;STR R2      ;LDN R2  ..CK IF RAM EXIST
82B6 FB5A3AAE;    0630          XRI#5A          ;BNZ STACK      ..BR IF NO RAM
82BA 650B;        0631  HOMDSK:OUT 5,#0B          ..CLEAR ERROR FLAGS
82BC 6401;        0632          OUT 4,#01          ..OUTPUT UNIT#00
82BE 6521;        0633          OUT 5,#21          ..LOAD U/S#
82C0 E26E;        0634          SEX R2          ;IMP 6          ..READ STATUS
82C2 FA20;        0635          ANI#20          ..CK DRIVE
82C4 E0;          0636          SEX R0
82C5 3ACF;        0637          BNZ UNIT2          ..BR IF DRIVE FAIL
82C7 650D;        0638          OUT 5,#0D          ..SEEK TRACK#00
82C9 E26E;        0639  UNIT1:SEX R2          ;INP 6          ..READ STATUS
82CB F6;          0640          SHR          ..CK FOR BUSY
82CC 33C9;        0641          BDF UNIT1          ..BR IF BUSY
82CE E0;          0642          SEX R0
82CF 650B;        0643  UNIT2:OUT 5,#0B          ..CLEAR ERROR FLAGS
82D1 6441;        0644          OUT 4,#41          ..OUTPUT UNIT#1
82D3 6521;        0645          OUT 5,#21          ..LOAD U/S#
82D5 E26E;        0646          SEX R2          ;INP 6          ..READ STATUS
82D7 FA20;        0647          ANI#20          ..CK DRIVE
82D9 E0;          0648          SEX R0
82DA 3AE4;        0649          BNZ EXIT2          ..BR IF DRIVE FAIL
82DC 650D;        0650          OUT 5,#0D          ..SEEK TRACK#00
82DE E26E;        0651  UNIT:SEX R2          ;INP 6          ..READ STATUS
82E0 F6;          0652          SHR          ..CK FOR BUSY
82E1 33DE;        0653          BDF UNIT          ..BR IF BUSY
82E3 E0;          0654          SEX R0
82E4 6101;        0655  EXIT2:OUT 1,#01          ..SELECT RCA I/O
82E6 ;            0656                      ..GROUP
82E6 E2;          0657          SEX R2
82E7 D3;          0658          SEP R3
82E8 ;            0659  .. THE FOLLOWING ROUTINE DOES (?R) COMMAND
82E8 ;            0660  ..
82E8 ;            0661  ..
82E8 ;            0662  ..
82E8 81;          0663  TYPER: GLO SWITCH          ..CK IF ?
82E9 C280E1;      0664          LBZ SYNERR          ..BR IF NOT ?
82EC F89CA3;      0665          LDI A.0(TYPE5D) ;PLO SUB      ..SUB IS POINTING
82EF ;            0666                      ..TO TYPE5D ROUTINE
82EF D30D;        0667          SEP SUB,#0D          ..TYPE CR
82F1 D30A;        0668          SEP SUB,#0A          ..TYPE LF
82F3 F802AD;      0669          LDI#02          ;PLO CTER          ..TYPE R0,R1
82F6 D358;        0670  TYPEX: SEP SUB,T'X'          ..TYPE X
82F8 D358;        0671          SEP SUB,T'X'          ..SINCE R0,R1 ARE
82FA D358;        0672          SEP SUB,T'X'          ..CLOBBED BY UT20
82FC D358;        0673          SEP SUB,T'X'          ..X=DON'T CARE
82FE D320;        0674          SEP SUB,#20          ..TYPE SPACE
8300 2D;          0675          DEC CTER
8301 8D;          0676          GLO CTER
8302 CA82F6;      0677          LBNZ TYPEX          ..BR TO TYPE R1
8305 F804A0;      0678          LDI#04          ;PLO PTER          ..TYPE R2,R3
8308 F88CB0;      0679          LDI#8C          ;PHI PTER          ..LOAD ADDRESS
830E F802AD;      0680          LDI#02          ;PLO CTER          ..LOAD CTER
830E 40BF;        0681  TYP2: LDA PTER ;PHI CHAR          ..PRINT 2 HEX DIGIT
8310 F8AEA3;      0682          LDI A.0(TYPE2) ;PLO SUB

```

8313 D3;	0683	SEP SUB	
8314 40BF;	0684	LDA PTER ;PHI CHAR	
8316 F8AEA3;	0685	LDI A.0(TYPE2) ;PLO SUB	
8319 D3;	0686	SEP SUB	
831A F89CA3;	0687	LDI A.0(TYPE5D) ;PLO SUB	
831D 80FB08;	0688	GLO PTER ;XRI#08	
8320 C6;	0689	LSNZ	
8321 D32C;	0690	SEP SUB,T','	
8323 D320;	0691	SEP SUB,#20	
8325 2D;	0692	DEC CTER	
8326 8D3ADE;	0693	GLO CTER ;BNZ TYPER2	
8329 D358;	0694	SEP SUB,T'X'	
832B D358;	0695	SEP SUB,T'X'	
832D F809AD;	0696	LDI#09 ;PLO PTER	
8330 F88CB0;	0697	LDI#8C ;PHI PTER	
8333 F816AD;	0698	LDI#16 ;PLO CTER	
8336 40BF;	0699	LDA PTER ;PHI CHAR	
8338 F8AEA3;	0700	LDI A.0(TYPE2) ;PLO SUB	
833E D3;	0701	SEP SUB	..TYPE OTHER TWO
833C D320;	0702	TSPCE: SEP SUB; ,#20	..SPACE
833E ;	0703	..	
833E 40BF;	0704	TLOOPX: LDA PTER ;PHI CHAR	..FETCH ONE BYTE
8340 ;	0705		..FOR TYPING
8340 F8AEA3;	0706	LDI A.0(TYPE2) ;PLO SUB	
8343 D3;	0707	SEP SUB	..TYPE 2 HEX
8344 2D;	0708	DEC CTER	
8345 8D;	0709	GLO CTER	
8346 3A4C;	0710	BNZ TL3A	..BRANCH NOT DONE
8348 9D;	0711	GHI CTER	
8349 C2803F;	0712	LBZ START	..BRANCH IF DONE
834C 80FB18;	0713	TL3A: GLO PTER ;XRI#18	..CK IF RC?
834F 3A53;	0714	BNZ TLX	
8351 D32C;	0715	SEP SUB,T','	
8353 80FA0F;	0716	TLX: GLO PTER ;ANI #0F	..PTER DIV BY 16?
8356 3A5E;	0717	BNZ TL2A	
8358 D30D;	0718	SEP SUB; ,#0D	..THEN CR
835A D30A;	0719	SEP SUB,#0A	..TYPE LF
835C 303E;	0720	BR TLOOPX	
835E F6;	0721	TL2A: SHR	..DIV BY 2?
835F 333E;	0722	BDF TLOOPX	..NO, LOOP BACK
8361 303C;	0723	BR TSPCE	..ELSE TYPE SPACE &
8363 ;	0724		..LOOP BACK
8363 ;	0725	.. STANDARD CALL ROUTINE	
8363 D3;	0726	EXITA:SEP R3	..R3 IS POINTING
8364 ;	0727		..TO FIRST INSTR.
8364 ;	0728		..IN SUBROUTINE
8364 E2;	0729	CALL:SEX R2	..POINT TO STACK
8365 96;	0730	GHI R6	..PUSH R6 ONTO
8366 73;	0731	STXD	..STACK TO PREPARE
8367 ;	0732		..IT FOR PONTING
8367 86;	0733	GLO R6	..TO ARGUMENTS,
8368 ;	0734		..AND DECREMENT
8368 73;	0735	STXD	..TO FREE LOCATION.
8369 93;	0736	GHI R3	..COPY R3 INTO R6
836A B6;	0737	PHI R6	..TO SAVE RETURN
836B ;	0738		..ADDRESS
836B 83;	0739	GLO R3	..SAVE THE RETURN
836C ;	0740		..ADDRESS
836C A6;	0741	PLO R6	..SAVE THE RETURN
836D ;	0742		..ADDRESS
836D 46;	0743	LDA R6	..LOAD THE ADDRESS
836E ;	0744		..OF SUBROUTINE
836E B3;	0745	PHI R3	..INTO R3

836F 46;	0746	LDA R6	..INTO R3
8370 A3;	0747	PLO R3	..INTO R3
8371 3063;	0748	BR EXITA	..BRANCH TO ENTRY
8373 ;	0749		..POINT
8373 ;	0750	.. STANDARD RETURN ROUTINE	
8373 D3;	0751	EXITR:SEP R3	..RETURN TO MAIN
8374 ;	0752		..PROGRAM
8374 96;	0753	RET:GHI R6	..COPY R6 INTO R3
8375 B3;	0754	PHI R3	..R3 CONTAINS THE
8376 ;	0755		..RETURN
8376 86;	0756	GLO R6	..ADDRESS
8377 A3;	0757	PLO R3	..ADDRESS
8378 E2;	0758	SEX R2	..POINT TO STACK
8379 12;	0759	INC R2	..POINT TO SAVED
837A ;	0760		..OLD R6
837A 72;	0761	LDXA	..RESTORE THE
837B ;	0762		..CONTENTS
837B A6;	0763	PLO R6	..OF R6
837C F0;	0764	LDX	..OF R6
837D B6;	0765	PHI R6	..OF R6
837E 9F;	0766	GHI RF	
837F C08373;	0767	LBR EXITR	..BRANCH TO ENTRY
8382 ;	0768		..POINT
8382 ;	0769	.. UT20 VECTOR TABLE	
8382 ;	0770	ORG#83F0	
83F0 C0824D;	0771	OSTRNG:LBR MSGE	
83F3 C08298;	0772	INIT1:LBR DSKG01	
83F6 C0829E;	0773	INIT2:LBR DSKG02	
83F9 C08281;	0774	GOUT20:LBR ENTER	
83FC C0816C;	0775	CKHEX: LBR CKHXE	
83FF ;	0776	..	
83FF ;	0777	END	
0000			

!M

0000 ;	0001	
0000 ;	0002	
0000 ;	0003	
0000 ;	0004	
0000 ;	0005	
0000 ;	0006	ORG #8400
8400 ;	0007	..THIS ROUTINE IS USED TO LOAD A PROGRAM
8400 ;	0008	..WRITTEN IN UT2 FORMAT FROM ICOM FDSK
8400 ;	0009	..INTO MEMORY. THIS PROGRAM STARTS ASKING
8400 ;	0010	..FOR THE TRACK# AND UNIT#.THESE
8400 ;	0011	..NUMBERS SHOULD BE ENTERED FROM
8400 ;	0012	..TERMINAL AS HEX DIGITS, THEN THE PROGRAM
8400 ;	0013	..SEEKS THE U/TR AND LOAD THE PGM
8400 ;	0014	IRX=#60
8400 ;	0015	TYPE2=#81AE
8400 ;	0016	PTER=#0C ..DCB(DATA CONTROL BLOCK) P
.0		



```

8400 ; 0017 ST=#02 ..STACK POINTER
8400 ; 0018 PC=#03 ..MAIN PROGRAM COUNTER
8400 ; 0019 ASL=#0D
8400 ; 0020 AUX=#0E
8400 ; 0021 ..I/O PARAMETERS
8400 ; 0022 READAH=#813B
8400 ; 0023 TYPE=#81A4
8400 ; 0024 OSTRNG=#83F0
8400 ; 0025 DSKG0=#83F6
8400 ; 0026 GOUT20=#83F9
8400 ; 0027 CKHEX=#83FC
8400 F809A3; 0028 LDI A.0(START-#06) ;PLO R3
8403 F884B3; 0029 LDI A.1(START) ;PHI R3
8406 C083F6; 0030 LBR DSKG0
8409 F81FA2; 0031 LDI#1F ;PLO R2
840C F88CB2; 0032 LDI#8C ;PHI R2
840F F800ADB0; 0033 START: LDI#00 ;PLO ASL ;PHI ASL ..CLEAR ASL
8413 BA73; 0034 PHI RA ;STXD ..CLEAR RA
8415 D483F00D0A5245; 0035 ASK: SEP R4 ,A(OSTRNG),#0D,#0A,T'READ?',#00
841C 41443F00; 0035
8420 D4813B; 0036 ASK1: SEP R4; ,A(READAH) ..READ A CHAR
8423 FB0D; 0037 XRI#0D ..CK FOR A CR
8425 3A20; 0038 BNZ ASK1
8427 D483F00A4C4F41; 0039 SEP R4,A(OSTRNG),#0A,T'LOADING',#00
842E 44494E4700; 0039
8433 E3; 0040 SEX R3
8434 6400; 0041 OUT 4,#00 ..OUTPUT U/S#00
8436 9D; 0042 GHI ASL
8437 323D; 0043 BZ CONTIN ..BRANCH IF U/S#00
8439 F840; 0044 LDI#40 ..UNIT#1
843B 6440; 0045 OUT 4,#40 ..OUTPUT U/S#40
843D 52; 0046 CONTIN: STR ST ..STORE U/S# IN DCB
843E 226521; 0047 DEC ST ;OUT 5,#21 ..LOAD THE U/S#
8441 650B; 0048 OUT 5,#0B ..CLEAR ERROR FLAG
8443 650D; 0049 OUT 5,#0D ..SEEK TRACK#00
8445 ; 0050 ..THE FOLLOWING ROUTINE CONVERTS A DECIMAL# I
.N R9.
8445 ; 0051 ..HEX AND STORE IT @DCB PTER
8445 E28D; 0052 CVY: SEX R2;GLO ASL
8447 FF10; 0053 SMI#10
8449 3B52; 0054 ENF RESULT
844B AD; 0055 PLO ASL
844C 9A; 0056 GHI RA
844D FCOA; 0057 ADI#0A
844F BA; 0058 PHI RA
8450 3045; 0059 ER CVY ..BRANCH IF NOT NEGATIVE
8452 8D; 0060 RESULT: GLO ASL
8453 52; 0061 STR ST
8454 9A; 0062 GHI RA
8455 F4; 0063 ADD
8456 73; 0064 STXD
8457 82AC; 0065 GLO ST ;PLO PTER
8459 92EC1C; 0066 GHI ST ;PHI PTER ;INC PTER
845C 1C1C; 0067 INC PTER ;INC PTER ..PTER @THE BYTE COUN
.
845E D48506; 0068 SEP R4,A(EWAIT) ..WAIT UNTIL DISK NOT BUSY
8461 D48573; 0069 READX: SEP R4; ,A(READ) ..READ 1 ASCII DIGIT
8464 ; 0070 ..FROM READ BUFFER->RF.1
8464 CB83F9; 0071 LBNF GOUT20 ..READ ERROR RESTART
8467 FB21; 0072 XRI#21 ..CK FOR !
8469 327A; 0073 BZ ISITM
846B FB05; 0074 XRI #05 ..CHECK FOR $
846D 32C6; 0075 BZ ISITU

```

```

846F FB37;      0076      XRI #37          ..CHECK FOR EOF(DC3)
8471 3A61;      0077      BNZ READX
8473 D483F000;  0078 DONE: SEP R4 ,A(OSTRNG) ,#00  ..TYPE NULL MESSAGE
. RESET DELAY PTR
8477 C083F9;    0079      LBR GOUT20          ..TRANSFER CONTROL TO UT2
.0
847A D48573;    0080      ISITM: SEP R4; ,A(READ)
847D CB83F9;    0081      LBNF GOUT20          ..READ ERROR RESTART
8480 FB4D;      0082      XRI#4D          ..CK FOR M
8482 3AE1;      0083      BNZ ERRORX          ..IF NOT M->ERROR
8484 D484F6;    0084      READX1:SEP R4; ,A(READHX)  ..READ 1 ASCII DIGIT
8487 ;          0085      ..AND CK IF HEX
8487 3399;      0086      BDF READX2          ..BR IF HEX
8489 FB2E;      0087      XRI#2E          ..CK IF "."
848B 3A84;      0088      BNZ READX1
848D D48573;    0089      READXA:SEP R4; ,A(READ)
8490 CB83F9;    0090      LBNF GOUT20          ..READ ERROR RESTART
8493 FB0D;      0091      XRI#0D
8495 3A8D;      0092      BNZ READXA
8497 3084;      0093      PR READX1
8499 D484F6;    0094      READX2:SEP R4; ,A(READHX)  ..READ 2ND ASCII DIGIT
.
849C ;          0095      ..AND CK IF HEX
849C 3399;      0096      BDF READX2          ..BR IF HEX
849E FB20;      0097      XRI#20          ..CK IF SPACE
84A0 3AE1;      0098      BNZ ERRORX          ..BR IF NOT SPACE
84A2 8DA8;      0099      GLO ASL ;PLO R8          ..ADDRESS->R8
84A4 9DB8;      0100      GHI ASL ;PHI R8
84A6 D484F6;    0101      READX3:SEP R4; ,A(READHX)  ..READ AN ASCII DIGIT
84A9 ;          0102      ..AND CK FOR HEX
84A9 3BBC;      0103      ENF READX4          ..BR IF NOT HEX
84AB D484F6;    0104      READXB:SEP R4; ,A(READHX)  ..READ THE 2ND ASCII D
.I
84AE ;          0105      ..AND CK IF HEX
84AE 3BE1;      0106      ENF ERRORX          ..BR IF NOT HEX->ERROR
84E0 8D58;      0107      GLO ASL ;STR R8          ..STORE AT THE SPECIFIED
.
84E2 ;          0108      ..ADDRESS
84E2 E8F3;      0109      SEX R8; XOR          ..DID IT WRITE CORRECTLY?
84E4 32B9;      0110      BZ WRTOK          ..YES
84E6 D48779;    0111      SEP R4 ,A(NOTRAM)          ..NO
84E9 18;        0112 WRTOK: INC R8
84EA 30A6;      0113      PR READX3
84EC FB0D;      0114      READX4:XRI#0D          ..CK IF CR
84EE 3261;      0115      BZ READX          ..IF CR->DONE
84F0 FB36;      0116      XRI#36          ..CK FOR SEMICOLON
84F2 ;          0117      ..TEST WITH(CR.XOR.,.XOR.
.
84F2 328D;      0118      BZ READXA          ..BR IF SEMICOLON
84F4 30A6;      0119      PR READX3
84F6 D48573;    0120      ISITU: SEP R4 ,A(READ)
84F9 CB83F9;    0121      LBNF GOUT20          ..READ ERROR RESTART
84FC FE55;      0122      XRI T'U
84FE 3AE1;      0123      BNZ ERRORX
8500 D484F6;    0124      ADLP: SEP R4 ,A(READHX)
8503 33D0;      0125      BDF ADLP
8505 8DA0;      0126      GLO ASL ;PLO R0
8507 9DB0;      0127      GHI ASL ;PHI R0
8509 D483F00D0A00; 0128      SEP R4; ,A(OSTRNG),#0D0A,#00          ..OUTPUT A
. LF
850F E0;        0129      SEX R0
8510 D0;        0130      SEP R0
8511 D483F00D0A464F; 0131      ERRORX: SEP R4; ,A(OSTRNG),#0D0A,T'FORMAT ERROR',#00
.
85E8 524D4154204552; 0131
85EF 524F5200;   0131

```

```

84F3 C083F9;      0132      LBR GOUT20
84F6 ;            0133      ..SUBROUTINES
84F6 ;            0134      ..THIS ROUTINE READS 1 ASCII DIGIT FROM DISK
84F6 D48573;      0135      READHX: SEP R4; ,A(READ)    ..READ 1 ASCII DIGIT
84F9 CB83F9;      0136      LBNF GOUT20      ..READ ERROR RESTART
84FC D483FC;      0137      SEP R4; ,A(CKHEX)    ..CK IF HEX
84FF D5;          0138      EXIT:  SEP R5
8500 ;            0139      ORG #8500
8500 ;            0140      ..
8500 ;            0141      .....BRANCH POINTS
8500 ;            0142      ..
8500 30C6;        0143      EWRITE: BR WRITE      ..ENTRY TO DISK WRITE ROUTINE

8502 3073;        0144      EREAD: BR READ    ..ENTRY TO DISK READ ROUTINE
8504 3017;        0145      ETRNFR: BR TRNFR1
8506 3011;        0146      EWAIT: BR WAIT1    ..ENTRY TO SIMPLE WAIT ROUTINE
8508 C08629;      0147      DER: LBR DERROR
850E C0860E;      0148      EWAITD: LBR WAIT
850E C087A0;      0149      LINEPR: LBR PRNTRF      ..LINE PRINTER UTILITY
8511 ;            0150      ..
8511 ;            0151      .....SUBROUTINE WAIT1
8511 ;            0152      ..
8511 E2;          0153      WAIT1:  SEX R2
8512 6E;          0154      INP 6      ..GET DISK STATUS
8513 F6;          0155      SHR      ..BUSY=>DF
8514 3311;        0156      BDF WAIT1
8516 D5;          0157      SEP R5      ..RETURN
8517 ;            0158      ..
8517 ;            0159      .....SUBROUTINE TRNFR1
8517 ;            0160      ..
8517 F810AF;      0161      TRNFR1: LDI #10; PLO RF      ..16 ERRORS ALLOWED
851A 4C52;        0162      LDA PTER; STR R2      ..GET TRK #,STORE ON
      .STACK
851C ;            0163      ..POINT TO UNIT-SECT #
851C 6422;        0164      OUT 4; DEC R2      ..OUTPUT THE TRACK #
851E E36511E2;    0165      SEX R3; OUT 5 ,#11; SEX R2      ..LOAD TRK #
8522 4C52;        0166      LDA PTER; STR R2      ..GET UNIT-SECT #,STO
      .RE ON STACK
8524 ;            0167      ..POINT TO BYTE COUNT
8524 6422;        0168      OUT 4; DEC R2      ..OUTPUT UNIT-SECT #
8526 E36521;      0169      SEX R3; OUT 5 ,#21      ..LOAD UNIT-SECT #
8529 C4C4C4C4;    0170      NOP; NOP; NOP      ..WAIT 48US FOR DISK
852D 6509;        0171      OUT 5 ,#09      ..SEEK TRACK
852F D4860E;      0172      SEP R4 ,A(WAIT)      ..WAIT TO SEEK
8532 9FFE;        0173      GHI RF;SHL      ..ERROR FLAG=>DF
8534 336E;        0174      BDF TRNEXT      ..DRIVE FAIL ERROR, RETURN
8536 6EFA08;      0175      INP 6; ANI #08      ..CHECK FOR CRC ERROR
8539 3241;        0176      BZ RDWTCK      ..NO CRC ERROR
853B 9FF940BF;    0177      GHI RF;ORI #40;PHI RF      ..SET SEEK ERROR FLAG

853F 306E;        0178      BR TRKNG      ..PRINT SEEK ERROR
8541 E3650B;      0179      RDWTCK: SEX R3; OUT 5 ,#0B      ..CLEAR ERROR FLAGS
8544 9F;          0180      GHI RF      ..CK READ/WRITE FLAG
8545 F6;          0181      SHR      ..FLAG INTO DF
8546 334C;        0182      BDF WRTCK      ..BR IF WRITE
8548 6503;        0183      OUT 5 ,#03      ..READ
854A 3054;        0184      BR STATUS      ..WAIT FOR READ
854C 6505;        0185      WRTCK:  OUT 5 ,#05      ..WRITE
854E D4860E;      0186      SEP R4 ,A(WAIT)      ..WAIT TO DO THE WRIT
      .E
8551 E36507;      0187      SEX R3; OUT 5 ,#07      ..READ CRC
8554 D4860E;      0188      STATUS: SEP R4 ,A(WAIT)      ..WAIT FOR COMMAND TO EXECUTE

8557 6E;          0189      INP 6
8558 FA08;        0190      ANI #08      ..BIT3=1=>CRC ERROR

```

```

855A 326E;          0191      BZ TRNEX          ..NO CRC ERROR RETURN
855C 2F;            0192      DEC RF             ..DEC THE ALLOWED ERROR COUNT

.
855D 8F;            0193      GLO RF             ..ANY MORE ALLOWED?
855E 3A41;          0194      BNZ RDWTCK         ..BR IF YES
8560 D48629;        0195      SEP R4,A(DERROR)
8563 9FFA01F920BF; 0196      GHI RF;ANI #01;ORI #20;PHI RF ..SET CRC FLA
.G
8569 306E;          0197      ER TRNEX          ..RETURN
856E D48629;        0198 TRNG: SEP R4,A(DERROR) ..PRINT ERROR
856E 9FFBFFFE;      0199 TRNEX: GHI RF;XRI #FF;SHL ..SET/RESET ERROR FLA
.G=>DF
8572 D5;            0200      SEP R5 ..RETURN
8573 ;              0201      ..
8573 ;              0202      .....SUBROUTINE READ
8573 ;              0203      ..
8573 EC;            0204 READ: SEX PTER
8574 F800BF;        0205      LDI #00;PHI RF ..SET READ MODE
8577 F0;            0206      LDX ..GET BYTE COUNT
8578 3AB0;          0207      BNZ SHFTBR ..BUFFER NOT EMPTY, SHIFT BUFFER

.
857A F800BF;        0208 REREAD: LDI #00;PHI RF ..SET READ MODE
857D F88073;        0209      LDI #80; STXD ..INITIALIZE BYTE COUNT FOR N
.EXT SECTOR
8580 ;              0210      ..POINT AT UNIT-SECT #
8580 F0FC0173;      0211      LDX; ADI #01; STXD ..INCR SECTOR #, POINT A
.T TRK #
8584 FA1F;          0212      ANI #1F ..MASK OUT UNITS BITS
8586 FD1A;          0213      SDI #1A ..SECTOR > 26 ?
8588 3395;          0214      PDF CNTOK ..NO, CHECK TRACK #
858A 1C;            0215      INC PTER ..POINT AT UNIT SECT #
858B F0FAC0FC0173; 0216      LDX; ANI #C0; ADI #01; STXD ..RESET SECT # T
.O 1, POINT TRK #
8591 F0FC015C;      0217      LDX; ADI #01; STR PTER ..INCR TRK #
8595 F0;            0218 CNTOK: LDX ..GET TRK #
8596 FD4C;          0219      SDI #4C ..TRK # > 76 ?
8598 33A1;          0220      PDF TRKOK ..TRACK IN RANGE, OK
859A 1C1C;          0221      INC PTER; INC PTER ..POINT BYTE COUNT
859C E36500;        0222 SEX R3;OUT 5,#00 ..SET UP TO READ STATUS
859F 306E;          0223      BR TRNG ..PRINT TRACK RANGE ERROR
85A1 E2;            0224 TRKOK: SEX R2
85A2 D48517;        0225      SEP R4,A(TRNFR1) ..READ A SECTOR
.FROM DISK TO BUFFER
85A5 3BBC;          0226      BNF RDXIT ..ERROR NOT CRC
85A7 EC;            0227      SEX PTER
85A8 9FFA203A7A;    0228      GHI RF;ANI #20;BNZ REREAD ..READ NEXT S
.ECTOR ON CRC ERR
85AD F840;          0229      LDI #40 ..EXAMINE READ BUFFER
85AF C8;            0230      LSKP
85B0 F841;          0231 SHFTBR: LDI #41 ..SHIFT READ BUFFER
85B2 52;            0232      STR R2
85B3 E2;            0233      SEX R2
85B4 6522;          0234      OUT 5; DEC R2
85B6 6E;            0235      INP 6 ..READ A BYTE
85B7 AF;            0236      PLO RF ..SAVE IT
85B8 0CFF015C;      0237      LDN PTER; SMI #01; STR PTER ..DEC BYTE COUNT

.
85EC 9FFBFFFE;      0238 RDXIT: GHI RF;XRI #FF;SHL ..SET/RESET ERROR FLA
.G=>DF
85C0 8F;            0239      GLO RF ..GET READ BYTE
85C1 CF;            0240      LSDF ..IF NO ERROR RETURN CHARACTER
85C2 F813;          0241      LDI #13 ..IF ERROR RETURN DC3
85C4 BF;            0242      PHI RF
85C5 D5;            0243      SEP R5 ..RETURN
85C6 ;              0244      ..

```

```

85C6 ; 0245 .....SUBROUTINE WRITE
85C6 ; 0246 ..
85C6 9F52; 0247 WRITE: GHI RF; STR R2 ..SAVE DATA BYTE TO STACK
85C8 6422; 0248 OUT 4; DEC R2 ..OUTPUT THE DATA
85CA F801BF; 0249 LDI #01; PHI RF ..SET WRITE MODE
85CD E36531; 0250 SEX R3; OUT 5 ,#31 ..LOAD WRITE BUFFER
85D0 EC; 0251 SEX PTER ..POINT TO BYTE COUNT
85D1 F0FC01; 0252 LDX; ADI #01 ..INC THE BYTE COUNT
85D4 5C; 0253 STR PTER
85D5 FF80; 0254 SMI #80 ..BYTE COUNT< 128 ?
85D7 CB8609; 0255 LBNF EXWT ..BR IF YES
85DA F801BF; 0256 REWRIT: LDI #01; PHI RF ..SET WRITE MODE
85DD F80073; 0257 LDI #00; STXD ..ZERO THE BYTE COUNT
85E0 ; 0258 ..POINT AT THE SEC#
85E0 F0FC0173; 0259 LDX; ADI #01; STXD ..INC SEC# AND POINT
.TRK#
85E4 FA1F; 0260 ANI #1F ..MASK OUT UNIT NUMBER
85E6 FD1A; 0261 SDI #1A ..SECTOR > 26 ?
85E8 33F5; 0262 BDF WTCNT ..NO, CHECK TRK #
85EA 1C; 0263 INC PTER ..POINT AT UNIT/SEC#
85EB F0FAC0FC0173; 0264 LDX; ANI #C0; ADI #01; STXD ..RESET SECT # TO
. 1, POINT TRK #
85F1 F0FC01; 0265 LDX; ADI #01 ..INC THE TRACK#
85F4 5C; 0266 STR PTER
85F5 F0; 0267 WTCNT: LDX .. GET THE TRK #
85F6 FD4C; 0268 SDI #4C ..TRK # > 76 ?
85F8 33FE; 0269 BDF TRKOK1 ..TRACK IN RANGE, OK
85FA 1C1C; 0270 INC PTER; INC PTER ..POINT TO BYTE COUNT
.
85FC 306E; 0271 BR TRKNG ..PRINT TRACK RANGE ERROR
85FE E2; 0272 TRKOK1: SEX R2
85FF D48517; 0273 SEP R4, A(TRNFR1) ..WRITE BUFFER TO DIS
.K
8602 EC; 0274 SEX PTER
8603 9FFA20CA85DA; 0275 GHI RF; ANI #20; LBNZ REWRIT ..CRC ERROR
.WRITE NEXT SECTOR
8609 9FFBFFFE; 0276 EXWT: GHI RF; XRI #FF; SHL ..SET/RESET ERROR FLA
.G=>DF
860D D5; 0277 SEP R5 ..RETURN
860E ; 0278 ..
860E ; 0279 ..
860E ; 0280 .....SUBROUTINE WAIT
860E ; 0281 ..
860E E2; 0282 WAIT: SEX R2
860F 6E; 0283 INP 6 ..GET STATUS
8610 FA20; 0284 ANI #20 ..DRIVE FAIL ?
8612 3A19; 0285 PNZ FAILUR ..DRIVE FAILED, PRINT ERROR
8614 6E; 0286 INP 6 ..GET STATUS
8615 FA40; 0287 ANI #40 ..DRIVE ACTIVE?
8617 3A1F; 0288 BNZ NOFAIL ..YES, NO FAILURE
8619 F800AF; 0289 FAILUR: LDI #00; PLO RF ..CLEAR TRY COUNT, DRIVE FAIL
.
861C D48629; 0290 SEP R4, A(DERROR) ..PRINT DRIVE FAILURE
.
861F 6E; 0291 NOFAIL: INP 6 ..GET STATUS
8620 FA08; 0292 ANI #08 ..CRC ERROR?
8622 3A28; 0293 PNZ RETWAT ..IF ERROR RETURN
8624 6EF6; 0294 INP 6; SHR ..CHECK IF OPERATION DONE
8626 330E; 0295 BDF WAIT ..NOT DONE
8628 D5; 0296 RETWAT: SEP R5 ..RETURN
8629 ; 0297 ..
8629 ; 0298 .....SUBROUTINE DISK ERROR
8629 ; 0299 ..
8629 E2; 0300 DERROR: SEX R2
862A 9C738C73; 0301 GHI RC; STXD; GLO RC; STXD ..SAVE DCB POINTERS

```



```

862E 9F738F73;      0302      GHI RF;STXD GLO RF;STXD ..SAVE FLAGS AND ERRO
.R COUNT
8632 9A738A73;      0303      GHI RA;STXD;GLO RA;STXD ..SAVE RA
8636 2C;             0304      DEC RC ..POINT TO UNIT-SECT#
8637 0CFA7FAA;      0305      LDN RC;ANI #7F;PLO RA ..SAVE UNIT-SECTOR# I
.N RA.0
863B 2C;             0306      DEC RC ..POINT TO TRK#
863C 0CBA;           0307      LDN RC;PHI RA ..SAVE TRACK# IN RA.1
863E 9F73;           0308      GHI RF;STXD ..SAVE FLAGS
8640 FA40;           0309      ANI #40 ..CHECK FOR SEEK ERROR
8642 325C;           0310      BZ ERR10 ..NOT SEEK ERROR
8644 D483F00D0A5452;0311      SEP R4,A(OSTRNG),#0DOA,T'TRACK SEEK ERROR',#0
.O
864B 41434E20534545;0311
8652 4E204552524F52;0311
8659 00;             0311
865A 30BE;           0312
865C 6EFA20;         0313 ERR10: INP 6;ANI #20 ..CHECK FOR DRIVE FAILURE
865F 3276;           0314      BZ ERR20 ..NOT DRIVE RFAILURE
8661 D483F00D0A4452;0315      SEP R4,A(OSTRNG),#0DOA,T'DRIVE FAILURE',#00
8668 49564520464149;0315
866F 4C55524500;     0315
8674 30BE;           0316
8676 6EFA08;         0317 ERR20: INP 6;ANI #08 ..CHECK FOR CRC ERROR
8679 3290;           0318      BZ ERR30 ..NOT CRC ERROR
867E D483F00D0A4352;0319      SEP R4,A(OSTRNG),#0DOA,T'CRC ERROR',#00
8682 43204552524F52;0319
8689 00;             0319
868A 8AF980AA;       0320
868E 30BE;           0321
8690 6EFA40;         0322 ERR30: INP 6;ANI #40 ..DRIVE ACTIVE?
8693 3AAF;           0323      BNZ ERR40 ..YES
8695 D483F00D0A4452;0324      SEP R4,A(OSTRNG),#0DOA,T'DRIVE NOT ACTIVE',#0
.O
869C 495645204E4F54;0324
86A3 20414354495645;0324
86AA 00;             0324
86AB 60;             0325
86AC C08738;         0326
86AF D483F00D0A5452;0327 ERR40: ,IRX
86B6 41434E203E3736;0327      LBR DERXT ..EXIT
86BD 00;             0327      SEP R4,A(OSTRNG),#0DOA,T'TRACK >76',#00
86BE 60F0;           0328 RDWRPT: ,IRX;LDX ..GET FLAGS
86C0 F6;             0329      SHR ..WRITE FLAG=>DF
86C1 33D5;           0330      EDF WROP ..PRINT "DURING WRITE"
86C3 D483F020445552;0331 RDOP: SEP R4,A(OSTRNG),T' DURING READ',#00
86CA 494E4720524541;0331
86D1 4400;           0331
86D3 30E6;           0332
86D5 D483F020445552;0333 WROP: BR UTSPT ..PRINT "UNIT TRACK SECTOR"
86DC 494E4720575249;0333      SEP R4,A(OSTRNG),T' DURING WRITE',#00
86E3 544500;         0333
86E6 D483F020554E49;0334 UTSPT: SEP R4,A(OSTRNG),T' UNIT ',#00
86ED 542000;         0334
86F0 8AFA40;         0335      GLO RA;ANI #40 ..GET UNIT-SECT # AND TEST UN
.IT BIT
86F3 CE;             0336      LSZ ..NOT UNIT 1
86F4 F801;           0337      LDI #01
86F6 FC30;           0338      ADI #30
86F8 EF;             0339      PHI RF
86F9 D481A4;         0340      SEP R4,A(TYPE) ..TYPE UNIT #
86FC D483F02C205452;0341      SEP R4,A(OSTRNG),T', TRACK ',#00
8703 41434B2000;     0341

```

```

8708 9ABF;          0342      GHI RA; PHI RF ..GET TRACK #
870A D4874D;        0343      SEP R4,A(TYPECD) ..TYPE TRACK #
870D D483F02C205345;0344      SEP R4,A(OSTRNG),T', SECTOR ',#00
8714 43544F522000; 0344
871A 8AFA1FBF;      0345      GLO RA;ANI #1F;PHI RF ..GET SECTOR #
871E D4874D;        0346      SEP R4,A(TYPECD) ..TYPE SECTOR #
8721 8AFA803232;    0347      GLO RA;ANI #80;PZ NOSKIP ..CHECK CRC F
.LAG
8726 D483F020534E49;0348      SEP R4,A(OSTRNG),T' SKIPPED',#00
872D 5050454400;    0348
8732 D483F00D0A00;  0349 NOSKIP: SEP R4,A(OSTRNG),#0D0A00 ..TYPE CR-LF
8738 60;            0350 DERXT: ,IRX ..RECOVER REGISTERS FROM STACK
8739 72AA72EA;      0351      LDXA;PLO RA;LDXA;PHI RA ..GET OLD RA
873D 72AF72;        0352      LDXA;PLO RF;LDXA ..GET FLAGS AND ERROR CO
.UNT
8740 FA01F980FF;    0353      ANI #01;ORI #80;PHI RF ..SET ERROR FLAG
8745 72ACF0EC;      0354      LDXA;PLO RC;LDX;PHI RC ..GET DCB POINTER
8749 E3650E;        0355      SEX R3;OUT 5,#0F ..CLEAR ERROR FLAGS
874C D5;            0356      SEP R5 ..RETURN
874D ;              0357 ..
874D ;              0358 .....SUPERROUTINE TO PRINT HEX IN RF.1 AS BCD #
874D ;              0359 ..
874D 9F;            0360 TYPECD: GHI RF ..GET INPUT
874E FAFO;          0361      ANI #FO ..STORE AS 2 HEX DIGITS
8750 F6F6F6F6;      0362      SHR;SHR;SHR;SHR
8754 AF;            0363      PLO RF
8755 9FFA0FBF;      0364      GHI RF;ANI #0F;PHI RF
8759 FFOA;          0365      SMI 10 ..DECIMAL ADJUST LOW DIGIT
875E 3E61;          0366      BNF SIXLP
875D 9FFC06BF;      0367      GHI RF;ADI 6;PHI RF
8761 8F;            0368 SIXLP: GLO RF ..ADD 16 TO BCD NUMBER FOR EACH HIGH
.HEX COUNT
8762 3275;          0369      PZ EXITBC ..IF HIGH COUNT=0, EXIT
8764 2F;            0370      DEC RF
8765 9FFC16BF;      0371      GHI RF;ADI #16;PHI RF
8769 FAOF;          0372      ANI #0F
876E FFOA;          0373      SMI 10 ..DECIMAL ADJUST BCD RESULT
876D 3E61;          0374      BNF SIXLP
876F 9FFC06BF;      0375      GHI RF;ADI 6;PHI RF
8773 3061;          0376      ER SIXLP ..LOOP UNTIL DONE
8775 D481AE;        0377 EXITBC: SEP R4,A(TYPE2)
8778 D5;            0378      SEP R5 ..RETURN
8779 ;              0379 .....
8779 8C73;          0380 NOTRAM: GLO RC;STXD
877B 9C73;          0381      GHI RC;STXD
877D D483F00D0A5241;0382      SEP R4,A(OSTRNG),#0D0A,T'RAM AT ',#00
8784 4D2041542000; 0382
878A 98BF;          0383      GHI R8;PHI RF
878C D481AE;        0384      SEP R4,A(TYPE2)
878F 88BF;          0385      GLO R8;PHI RF
8791 D481AE;        0386      SEP R4,A(TYPE2)
8794 D483F0203F00; 0387      SEP R4,A(OSTRNG),T' ?',#00
879A 12;            0388      INC R2
879E 42EC;          0389      LDA R2;PHI RC
879D 02AC;          0390      LDN R2;PLO RC
879F D5;            0391      SEP R5
87A0 ;              0392 .....
87A0 ;              0393 ..THIS ROUTINE PRINTS TO THE LINE PRINTER, THE CONTEN
.TS OF RF.1.
87A0 ;              0394 ..IT SUPRESSES PRINTING OF LINE FEEDS, AND REPLACES C
.ARRIAGE RETURNS
87A0 ;              0395 ..WITH A CR-LF PAIR.

```

87A0 ;  
.BUT IF THE  
87A0 ;  
. WILL BE RESET  
87A0 ;  
87A0 ;  
87A0 ;  
87A0 9FFB0A;  
87A3 32BC;  
87A5 9FFB13;  
87A8 32C0;  
87AA 9FFBFF52;  
87AE 34AE;  
87E0 6622;  
87E2 9FFB0D;  
87E5 3ABC;  
87E7 F80ABF;  
87BA 30AA;  
87BC F801F6;  
87BF D5;  
87C0 F6;  
87C1 D5;  
87C2 ;  
0000

0396 ..NORMALLY, THIS ROUTINE RETURNS WITH THE DFLAG SET,  
0397 ..CHARACTER IN RF.1 WAS A DC3(END OF FILE), THE DFLAG  
0398 ..ON RETURN.  
0399 ..  
0400 ..  
0401 PRNTRF: GHI RF;XRI #0A ..IF LINE FEED, EXIT  
0402 BZ EXITDF  
0403 GHI RF;XRI #13 ..IF DC3, EXIT  
0404 BZ EXITEF  
0405 PRINT1: GHI RF;XRI #FF;STR R2 ..INVERT DATA  
0406 R1 \* ..WAIT UNTIL READY  
0407 OUT 6; DEC R2 ..OUTPUT CHARACTER  
0408 GHI RF;XRI #0D ..CARRIAGE RETURN?  
0409 BNZ EXITDF ..NO, EXIT  
0410 LDI #0A;PHI RF ..YES, PRINT A LINE FEED  
0411 BR PRINT1  
0412 EXITDF: LDI #01;SHR ..SET DFLAG  
0413 SEP R5 ..AND RETURN  
0414 EXITEF: SHR ..RESET DFLAG  
0415 SEP R5 ..AND RETURN  
0416 END