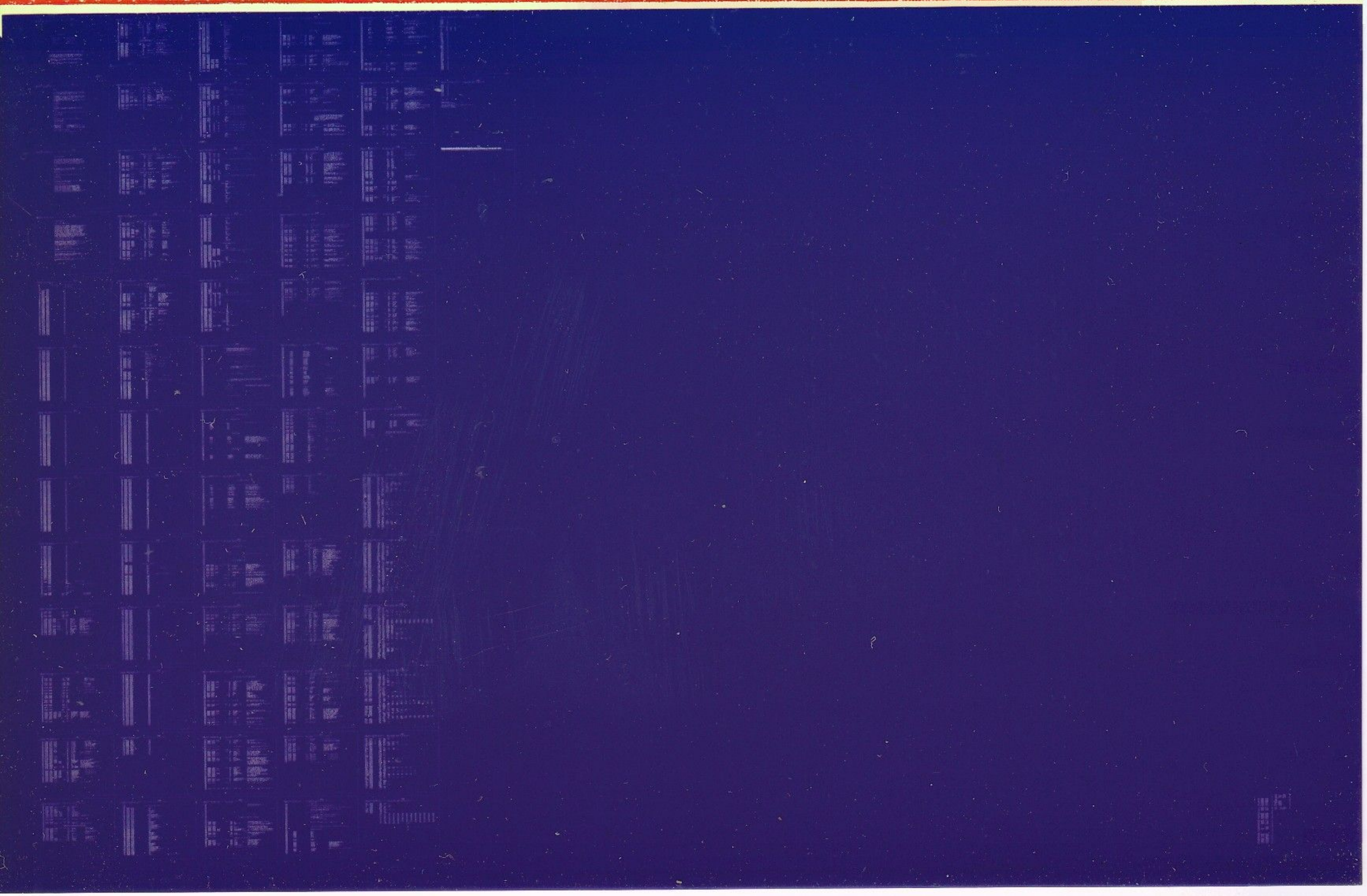
GT40

QUICK VERIFY
MD-11-DDGTE-C

EP-DDGTE-C-DL-B

FICHE 1 OF 1

DEC 1976
digital
MADE IN USA



```
.REM  +
```

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DDGTE-C-D
PRODUCT NAME:	GT40 QUICK VERIFY
DATE:	DECEMBER 1976
MAINTAINER:	DIAGNOSTIC GROUP

1. ABSTRACT

THIS VERSION OF THE PROGRAM SUPPORTS NON-SWITCH REGISTER CPU'S. FOR THESE CPU'S, THE SWITCH REGISTER CAN BE CHANGED BY CHANGING THE CONTENTS OF SWREG (170).

THIS PROGRAM IS A QUICK GO-NOGO TEST OF THE GT40 SYSTEM. THE PURPOSE OF THIS TEST IS TO QUICKLY IDENTIFY ANY PROBLEM IN THE SYSTEM. THE PROGRAM WILL START THE DISPLAY AND THEN INITIATE THE COMMUNICATION LINE.
TWO BACKGROUND TASKS ARE EXECUTED, THE FIRST IS A GT-40 ROM VERIFY TEST. THE SECOND TASK IS A WORSE CASE NOISE TEST THRU MEMORY.

2. REQUIREMENTS

2.1 EQUIPTMENT

GT40 SYSTEM (11/05, DISPLAY PROCESSOR AND VR14 SCOPE)
MODEM TEST CONNECTOR WHICH CONNECTS DATA OUT TO DATA IN.

2.2 STORAGE

THIS PROGRAM USED MEMORY LOCATIONS 0-7776 AND 16000-16776 (2K OF MEMORY).

3. LOADING PROCEDURE

3.1 METHOD

PROCEDURE FOR NORMAL BINARY TAPES SHOULD BE FOLLOWED.

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

```
CONSOLE SW 09 = 0       ROM PRESENT AS SPECIFIED BY SW 08
CONSOLE SW 09 = 1       NO ROM PRESENT
CONSOLE SW 08 = 0       TEST AS VERSION 2 ROM (512. WORDS)
CONSOLE SW 08 = 1       TEST AS VERSION 1 ROM (256. WORDS)
```

4.2 STARTING ADDRESS OR ADDRESSES

200 IS THE ONLY STARTING ADDRESS OF THIS TEST

5. OPERATING PROCEDURE

ONCE STARTED THE TEST WILL RUN IN THEIR NORMAL MANNER
WITHOUT OPERATOR INTERVENTION OR SWITCH SELECTION.
THE OPERATOR MUST VERIFY THE DATA RETURNING FROM THE COM-
MUNICATION LINE BY COMPARING 'COM OUTPUT' TO 'COM INPUT'
ON THE DISPLAY SCREEN. BY TYPING ON THE CONSOLE KEYBOARD,
THE CHARACTER AND OCTAL VALUE WILL BE DISPLAYED.

6. ERRORS

THE PROGRAM WILL ONLY HALT ON AN ERROR.
THE PROGRAM DOES NOT CONTAIN FACILITES FOR REPORTING
MESSAGES OR ERROR CONDITIONS.

7. RESTRICTIONS

A COMMUNICATION TEST PLUG MUST BE INSTALLED ON THE DL-11.

8. MISCELANEOUS

8.1 EXECUTION TIME

THE TEST WILL TAKE APPROXIMATELY 10 SECONDS FOR COMPLETION
AND WILL RING THE 'GT-40' BELL.

8.2 DEVICE ADDRESS PROGRAM LOCATIONS

LOCATION 1000 CONTAINS THE GT40 DEVICE ADDRESS
LOCATION 1002 CONTAINS THE GT40 INTERRUPT VECTOR.
LOCATION 1004 CONTAINS THE GT40 INTERRUPT LEVEL.
LOCATION 1006 CONTAINS THE DL-11 DEVICE ADDRESS.
LOCATION 1010 CONTAINS THE DL-11 INTERRUPT VECTOR.
LOCATION 1012 CONTAINS THE DL-11 INTERRUPT LEVEL.
LOCATION 1014 CONTAINS THE GT-40 ROM BOOTSTRAP ADDRESS.
LOCATION 1016 CONTAINS THE GT-40 ROM WORD LENGTH.

9. PROGRAM DESCRIPTION

9.1 DISPLAY FILE (FORGROUND TASK)

THE DISPLAY FILE IS A COMPACT VISUAL TEST OF ALL GT40 DISPLAY INSTRUCTIONS. A BOX OUTLINING THE SCREEN WITH DIFFERENT LINE TYPE VALUE IS DISPLAYED. THREE PAIRS OF ASCII STRINGS ARE ALSO DISPLAYED TO TEST THE CHARACTER LOGIC. THE FIRST LINE OF A STRING IS DISPLAYED IN 'NORMAL' FONT THE SECOND LINE OF A STRING IS DISPLAYED IN 'ITALICS'. ALSO INCLUDED IN THIS VISUAL TEST ARE THE 8 DIFFERENT INTENSITY LEVELS. THE DISPLAY PATTERN IS ENHANSED BY THE USE OF BLINKING OCTAGONS AND MOVING SINE WAVES. THE DISPLAY PATTERN ALSO SERVES AS FOR VISUAL INSPECTION OF THE COMMUNICATION LINE DATA. ALL LINES AND CHARACTERS ARE ENABLED FOR LIGHT-PEN INTERACTION EXCEPT FOR THE LARGEST OCTAGON. UPON LIGHT-PEN HIT, THE TEXT 'LIGHT-PEN HIT' WILL BE DISPLAYED NEAR CENTER SCREEN.

9.2 COMMUNICATION DATA (FORGROUND TASK)

THE DATA PRESENTED TO THE COMMUNICATION LINE APPEARS ON THE DISPLAY SCREEN AS FOUND AT 'COM OUTPUT'. (DECGRAPHIC-11 DISPLAY TERMINAL GT40 VR14) THE DATA ECHOED BACK BY THE TEST CONNECTOR IS DISPLAYED ON THE SCREEN AS FOUND AT 'COM INPUT'. A VISUAL TEST OF THE DATA MUST BE PERFORMED.

9.3 ROM VERIFY TEST (BACKGROUND TASK)

THIS TEST VERIFIES THE DATA CONTAINED IN THE GT-40 ROM BOOTSTRAP.

9.4 WORSE CASE NOISE TEST (BACKGROUND TASK)

THIS IS A BACKGROUND TEST OF ALL AVAILABLE MEMORY. A SMALL PROGRAM IS LOADED INTO ALL EXISTING MEMORY AND THEN EXECUTED THRU THE REMAINDER OF MEMORY.

9.5 KEYBOARD DATA (FORGROUND TASK)

UPON DEPRESSING A KEYBOARD KEY, THE OCTAL VALUE WILL BE DISPLAYED AND ECHO ONTO THE SCREEN.

```
                        .TITLE  GT40 QUICK VERIFY MAINDEC-11-DDGTE-C
                        .ENABL  ABS,AMA
                                                .LIST   SEQ,BIN
                                                .NLIST  MD,MC,CND

                000000                          .=C             ;0-776 IS LOADED WITH .+2, HALT
000000  000000                                  HALT
000002  000000                                  HALT
000004  000006                                  .+2
000006  000000                                  HALT
000010  000012                                  .+2
000012  000000                                  HALT
000014  000016                                  .+2
000016  000000                                  HALT
000020  000022                                  .+2
000022  000000                                  HALT
000024  000026                                  .+2
000026  000000                                  HALT
000030  000032                                  .+2
000032  000000                                  HALT
000034  000036                                  .+2
000036  000000                                  HALT
000040  000042                                  .+2
000042  000000                                  HALT
000044  000046                                  .+2
000046  000000                                  HALT
000050  000052                                  .+2
000052  000000                                  HALT
000054  000056                                  .+2
000056  000000                                  HALT
000060  000062                                  .+2
000062  000000                                  HALT
000064  000066                                  .+2
000066  000000                                  HALT
000070  000072                                  .+2
000072  000000                                  HALT
000074  000076                                  .+2
000076  000000                                  HALT
000100  000102                                  .+2
000102  000000                                  HALT
000104  000106                                  .+2
000106  000000                                  HALT
000110  000112                                  .+2
000112  000000                                  HALT
000114  000116                                  .+2
000116  000000                                  HALT
000120  000122                                  .+2
000122  000000                                  HALT
000124  000126                                  .+2
000126  000000                                  HALT
000130  000132                                  .+2
000132  000000                                  HALT
000134  000136                                  .+2
000136  000000                                  HALT
000140  000142                                  .+2
000142  000000                                  HALT
```

```
239  000144  000146                    .+2
240  000146  000000                    HALT
241  000150  000152                    .+2
242  000152  000000                    HALT
243  000154  000156                    .+2
244  000156  000000                    HALT
245  000160  000162                    .+2
246  000162  000000                    HALT
247  000164  000166                    .+2
248  000166  000000                    HALT
249  000170  000172                    .+2
250  000172  000000                    HALT
251  000174  000176                    .+2
252  000176  000000                    HALT
253  000200  000202                    .+2
254  000202  000000                    HALT
255  000204  000206                    .+2
256  000206  000000                    HALT
257  000210  000212                    .+2
258  000212  000000                    HALT
259  000214  000216                    .+2
260  000216  000000                    HALT
261  000220  000222                    .+2
262  000222  000000                    HALT
263  000224  000226                    .+2
264  000226  000000                    HALT
265  000230  000232                    .+2
266  000232  000000                    HALT
267  000234  000236                    .+2
268  000236  000000                    HALT
269  000240  000242                    .+2
270  000242  000000                    HALT
271  000244  000246                    .+2
272  000246  000000                    HALT
273  000250  000252                    .+2
274  000252  000000                    HALT
275  000254  000256                    .+2
276  000256  000000                    HALT
277  000260  000262                    .+2
278  000262  000000                    HALT
279  000264  000266                    .+2
280  000266  000000                    HALT
281  000270  000272                    .+2
282  000272  000000                    HALT
283  000274  000276                    .+2
284  000276  000000                    HALT
285  000300  000302                    .+2
286  000302  000000                    HALT
287  000304  000306                    .+2
288  000306  000000                    HALT
289  000310  000312                    .+2
290  000312  000000                    HALT
291  000314  000316                    .+2
292  000316  000000                    HALT
293  000320  000322                    .+2
294  000322  000000                    HALT
```

```
  295  000324  000326                    .+2
  296  000326  000000                    HALT
  297  000330  000332                    .+2
  298  000332  000000                    HALT
  299  000334  000336                    .+2
  300  000336  000000                    HALT
  301  000340  000342                    .+2
  302  000342  000000                    HALT
  303  000344  000346                    .+2
  304  000346  000000                    HALT
  305  000350  000352                    .+2
  306  000352  000000                    HALT
  307  000354  000356                    .+2
  308  000356  000000                    HALT
  309  000360  000362                    .+2
  310  000362  000000                    HALT
  311  000364  000366                    .+2
  312  000366  000000                    HALT
  313  000370  000372                    .+2
  314  000372  000000                    HALT
  315  000374  000376                    .+2
  316  000376  000000                    HALT
  317  000400  000402                    .+2
  318  000402  000000                    HALT
  319  000404  000406                    .+2
  320  000406  000000                    HALT
  321  000410  000412                    .+2
  322  000412  000000                    HALT
  323  000414  000416                    .+2
  324  000416  000000                    HALT
  325  000420  000422                    .+2
  326  000422  000000                    HALT
  327  000424  000426                    .+2
  328  000426  000000                    HALT
  329  000430  000432                    .+2
  330  000432  000000                    HALT
  331  000434  000436                    .+2
  332  000436  000000                    HALT
  333  000440  000442                    .+2
  334  000442  000000                    HALT
  335  000444  000446                    .+2
  336  000446  000000                    HALT
  337  000450  000452                    .+2
  338  000452  000000                    HALT
  339  000454  000456                    .+2
  340  000456  000000                    HALT
  341  000460  000462                    .+2
  342  000462  000000                    HALT
  343  000464  000466                    .+2
  344  000466  000000                    HALT
  345  000470  000472                    .+2
  346  000472  000000                    HALT
  347  000474  000476                    .+2
  348  000476  000000                    HALT
  349  000500  000502                    .+2
  350  000502  000000                    HALT
```

```
000504  000506                  .+2
000506  000000                  HALT
000510  000512                  .+2
000512  000000                  HALT
000514  000516                  .+2
000516  000000                  HALT
000520  000522                  .+2
000522  000000                  HALT
000524  000526                  .+2
000526  000000                  HALT
000530  000532                  .+2
000532  000000                  HALT
000534  000536                  .+2
000536  000000                  HALT
000540  000542                  .+2
000542  000000                  HALT
000544  000546                  .+2
000546  000000                  HALT
000550  000552                  .+2
000552  000000                  HALT
000554  000556                  .+2
000556  000000                  HALT
000560  000562                  .+2
000562  000000                  HALT
000564  000566                  .+2
000566  000000                  HALT
000570  000572                  .+2
000572  000000                  HALT
000574  000576                  .+2
000576  000000                  HALT
000600  000602                  .+2
000602  000000                  HALT
000604  000606                  .+2
000606  000000                  HALT
000610  000612                  .+2
000612  000000                  HALT
000614  000616                  .+2
000616  000000                  HALT
000620  000622                  .+2
000622  000000                  HALT
000624  000626                  .+2
000626  000000                  HALT
000630  000632                  .+2
000632  000000                  HALT
000634  000636                  .+2
000636  000000                  HALT
000640  000642                  .+2
000642  000000                  HALT
000644  000646                  .+2
000646  000000                  HALT
000650  000652                  .+2
000652  000000                  HALT
000654  000656                  .+2
000656  000000                  HALT
000660  000662                  .+2
000662  000000                  HALT
```

```
000664  000666                  .+2
000666  000000                  HALT
000670  000672                  .+2
000672  000000                  HALT
000674  000676                  .+2
000676  000000                  HALT
000700  000702                  .+2
000702  000000                  HALT
000704  000706                  .+2
000706  000000                  HALT
000710  000712                  .+2
000712  000000                  HALT
000714  000716                  .+2
000716  000000                  HALT
000720  000722                  .+2
000722  000000                  HALT
000724  000726                  .+2
000726  000000                  HALT
000730  000732                  .+2
000732  000000                  HALT
000734  000736                  .+2
000736  000000                  HALT
000740  000742                  .+2
000742  000000                  HALT
000744  000746                  .+2
000746  000000                  HALT
000750  000752                  .+2
000752  000000                  HALT
000754  000756                  .+2
000756  000000                  HALT
000760  000762                  .+2
000762  000000                  HALT
000764  000766                  .+2
000766  000000                  HALT
000770  000772                  .+2
000772  000000                  HALT
000774  000776                  .+2
000776  000000                  HALT
        000024                  .=24
000024  001674                  PWRFL           ;POWER FAIL VECTOR
000026  000340                  340
        000004                  ERRVEC=4
        177570                  DSWR=177570             ;11/45 LIGHT DISPLAY REGISTER

        000170                  .=170
000170  000000                  SWREG:  .WORD   0

        000200                  .=200

000200  000137  001022          JMP     STARTB

        001000                  .=1000
001000  172000          GTADD:  172000                  ;GT-40 ADDRESS
001002  000320          GTVCT:  320                     ;GT-40 VECTOR
001004  000200          GTBRL:  200                     ;GT-40 BR LEVEL
```

```
001006  175610                  DLADD:  175610                          ;DL-11 ADDRESS
001010  000300                  DLVCT:  300                             ;DL-11 VECTOR
001012  000240                  DLBRL:  240
001014  166000                  ROMADD: 166000                          ;ROM STARTING ADDRESS
001016  001000                  WORDS:  512.
001020  006000                  IMAGE:  6000
001022  012706  000500          STARTB: MOV     #500,SP         ;LOAD THE STACK POINTER
001026  012777  000340  000176          MOV     #340,@PSW       ;RAISE PSW
001034  004737  001316                  JSR     PC,INITGT       ;INIT DEVICE ADDRESSES
001040  005037  001020                  CLR     IMAGE           ;PRESET FOR NO ROM SELECTED
001044  032777  001000  000156          BIT     #1000,@SWR      ;TEST FOR ROM SELECTED SWITCH
001052  001021                          BNE     2$              ;NO ROM SELECTED
001054  032777  000400  000146          BIT     #400,@SWR       ;TEST ROM SWITCH
001062  001007                          BNE     1$              ;BR IF SET
001064  012737  001000  001016          MOV     #512.,WORDS     ;ASSUME VER. 2 ROM
001072  012737  006000  001020          MOV     #START,IMAGE    ;LOAD IMAGE ADDRESS
001100  000406                          BR      2$              ;START TEST
001102  012737  000400  001016  1$:     MOV     #256.,WORDS     ;SELECT VER. 1 ROM
001110  012737  016000  001020          MOV     #STARTA,IMAGE   ;LOAD IMAGE ADDRESS
001116  005077  000042          2$:     CLR     @DLODBR         ;CLEAR OUTPUT
001122  005077  000036                  CLR     @DLODBR
001126  004737  001720                  JSR     PC,DOCORE       ;SET UP CORE SIZE
001132  004737  001234                  JSR     PC,PRIME        ;INIT THE DEVICES
001136  005077  000070                  CLR     @PSW
001142  000137  002630                  JMP     OVER            ;EXECUTE BACKGROUND TASK
```

```
                                                GTPC:   172000          ;DISPLAY PC
001146  172000                                 GTSP:   172002          ;DISPLAY STATUS REG.
001150  172002                                 GTXPOS: 172004          ;DISPLAY X REGISTER
001152  172004                                 GTYPOS: 172006          ;DISSPLAY Y REGISTER
001154  172006

001156  175610                                 DLICSR: 175610      ;DL-11 STATUS
001160  175612                                 DLIDBR: 175612          ;DL-11 BUFFER
001162  175614                                 DLOCSR: 175614          ;DL-11 STATUS
001164  175616                                 DLODBR: 175616          ;DL-11 BUFFER

001166  000320                                 GTDONE: 320             ;DISPLAY DONE VECTOR
001170  000322                                 GTDNE1: 322

001172  000324                                 GTLPH:  324             ;DISPLAY LIGHT-PEN VECTOR
001174  000326                                 GTLPH1: 326

001176  000330                                 GTSOTM: 330             ;DISPLAY SHIFT-OUT/ TIME-OUT VECTOR
001200  000332                                 GTSOT1: 332

001202  000300                                 DLIVT:  300
001204  000302                                 DLIVT1: 302
001206  000304                                 DLOVT:  304
001210  000306                                 DLOVT1: 306

001212  177560                                 TKS:    177560
001214  177562                                 TKB:    177562
001216  177564                                 TPS:    177564
001220  177566                                 TPB:    177566

001222  000060                                 KRBVT:  60
001224  000062                                 KRBVT1: 62

001226  000200                                 KRBBRL: 200

001230  177570                                 SWR:    177570
001232  177776                                 PSW:    177776

001234  012777  003012  177704  PRIME:  MOV     #FILE00,@GTPC   ;START THE DISPLAY
001242  012777  000100  177712          MOV     #100,@DLOCSR    ;ENABLE DL OUTPUT
001250  012777  000100  177700          MOV     #100,@DLICSR    ;ENABLE DL INPUT
001256  012777  000100  177726          MOV     #100,@TKS       ;ENABLE KEYBOARD
001264  113777  005422  177672          MOVB    BUFF1,@DLODBR   ;OUTPUT A CHAR
001272  012737  000001  002430          MOV     #1,PPNT         ;PRESET PRINT POINTER
001300  005037  002432                  CLR     RPNT            ;CLEAR READ BUFFER
001304  005037  002624                  CLR     KPNT
001310  000207                          RTS     PC              ;EXIT

001312  017476                  SIZE:   17476
001314  000000                  GTDLYD: 0
```

```
001316  012700  001146          INITGT: MOV     #GTPC,R0                ;LOAD STARTING ADDRESS
001322  013701  001000                  MOV     GTADD,R1                ;SAVE VALUE
001326  004737  001400                  JSR     PC,LOADR0               ;LOAD GT ADDR
001332  013701  001006                  MOV     DLADD,R1                ;LOAD STARTING ADDRESS (DL-11)
001336  004737  001400                  JSR     PC,LOADR0               ;LOAD DL-11 ADDRESSES
001342  013701  001002                  MOV     GTVCT,R1                ;LOAD VECTOR VALUE
001346  004737  001400                  JSR     PC,LOADR0               ;LOAD GT-40 VECTORS
001352  010110                          MOV     R1,(R0)
001354  062720  000010                  ADD     #10,(R0)+               ;LOAD GT TIME-OUT
001360  010110                          MOV     R1,(R0)
001362  062720  000012                  ADD     #12,(R0)+
001366  013701  001010                  MOV     DLVCT,R1                ;LOAD VECTOR VALUE
001372  004737  001400                  JSR     PC,LOADR0       ;LOAD DL-11 VECTORS
001376  000436                          BR      INGT            ;BR
001400  010120          LOADR0: MOV     R1,(R0)+                ;LOAD DONE
001402  010110                  MOV     R1,(R0)
001404  062720  000002                  ADD     #2,(R0)+
001410  010110                          MOV     R1,(R0)
001412  062720  000004                  ADD     #4,(R0)+                ;LOAD DONE
001416  010110                          MOV     R1,(R0)
001420  062720  000006                  ADD     #6,(R0)+                        ;LOAD PSW
001424  013746  000004                  MOV     @#ERRVEC,-(SP)  ;SAVE VECTORS CONTENTS
001430  012737  001456  000004          MOV     #1$,@#ERRVEC    ;SET UP FOR TRAP
001436  012737  177570  001230          MOV     #DSWR,@#SWR     ;SET UP TO TEST FOR SWITCH REGISTER
001444  022777  177777  177556          CMP     #-1,@SWR        ;TEST FOR SWITCH REGISTER
001452  001005                          BNE     3$              ;SWITCH REGISTER IS PRESENT
001454  000401                          BR      2$              ;NO SWITCH REGISTER
001456  022626                  1$:     CMP     (SP)+,(SP)+     ;POP 2 WORDS OFF STACK
001460  012737  000170  001230  2$:     MOV     #SWREG,@#SWR    ;SET UP FOR SOFTWARE SWITCH REGISTER
001466  012637  000004          3$:     MOV     (SP)+,@#ERRVEC  ;RESTORE VECTORS CONTENTS
001472  000207                          RTS     PC                      ;EXIT

001474  012777  002130  177464  INGT:   MOV     #GTSTOP,@GTDONE         ;LOAD DONE VECTOR
001502  013777  001004  177460          MOV     GTBRL,@GTDNE1
001510  012777  002220  177454          MOV     #GTLPEN,@GTLPH          ;LOAD LIGHT-PEN VECTOR
001516  013777  001004  177450          MOV     GTBRL,@GTLPH1
001524  012777  002236  177444          MOV     #GTSHIF,@GTSOTM         ;LOAD SHIFT-OUT VECTOR
001532  013777  001004  177440          MOV     GTBRL,@GTSOT1
001540  012737  000040  001314          MOV     #40,GTDLY0
001546  012737  005752  005722          MOV     #FILEOC,FILEOA
001554  012737  174104  003074          MOV     #STATSB!INCR+4,GRPINC
001562  012700  005516                  MOV     #BUFF2,R0
001566  005020          INTD:   CLR     (R0)+
001570  022700  005566                  CMP     #BUFF2+50,R0
001574  001374                          BNE     INTD
001576  012700  005612                  MOV     #BUFF3,R0       ;SET UP KRB BUFFER
```

```
                                INTE:   CLR     (R0)+
                                        CMP     #BUFF3+50,R0
                                        BNE     INTE
                                        CLRB    OCTA
                                        CLRB    OCTA+1
                                        CLRB    OCTA+2
                                        MOV     #DLIN,@DLIVT
                                        MOV     DLBRL,@DLIVT1
                                        MOV     #DLOUT,@DLOVT
                                        MOV     DLBRL,@DLOVT1
                                        MOV     #KBIN,@KRBVT    ;LOAD KRB VECTOR
                                        MOV     KRBBRL,@KRBVT1  ;LOAD PSW
                                        RTS     PC

                                PWRFL:  MOV     #PWRUP,@#24     ;LOAD VECTOR
                                        HALT

                                PWRUP:  RESET
                                        MOV     #PWRFL,@#24
                                        JMP     STARTB          ;RESTART AT BEGINING

                                ;SUBROUTINE TO DETERMINE THE SIZE OF CORE
                                ; AND SET UP LOCATION SIZE WITH THE VALUE

                                DOCORE: MOV     #2$,@#4                 ;SET UP FOR NEM
                                        MOV     #17776,R1               ;SET UP ADDRESS
                                        CLR     R0
                                1$:     ADD     #20000,R1               ;MOVE TO THE NEXT BANK
                                        INC     R0                      ;INC BANK COUNTER
                                        TST     (1)                     ;TIMEOUT ?
                                        CMP     #157776,R1              ;END ?
                                        BNE     1$
                                        BR      3$
                                2$:     CMP     (SP)+,(SP)+             ;POP THE STACK X2
                                        DEC     R0                      ;DECREMENT BANK COUNT
                                        SUB     #20000,R1               ;RESTORE R1
                                3$:     MOV     #6,@#4                  ;RESET BUSS ERROR
                                        MOV     R1,SIZE
                                        SUB     #7776,SIZE              ;BACK PAST LOADER
```

```
                                        ;ROUTINE TO LOOK ACCESS CORE WITH WORSE CASE MEMORY TEST
                                
002004  013700  001312  XMTST:  MOV     SIZE,R0         ;GET LAST FREE CORE ADDRESS
002010  012701  001000          MOV     #BOOTEB,R1      ;GET END OF PROGRAM
002014  020001                  CMP     R0,R1           ;TEST FOR EQUAL
002016  103410                  BLOS    XMRTS           ;BRANCH IF NO ROOM
002020  012702  002054  XMLOP1: MOV     #MEMTST,R2      ;MOVE CODE BETWEEN
002024  012221          XMLOP2: MOV     (R2)+,(R1)+     ;MEMTST AND MEMEND TILL
002026  022702  002124          CMP     #MEMEND,R2      ;CORE IS FULL
002032  001374                  BNE     XMLOP2
002034  020100                  CMP     R1,R0           ;TEST FOR MORE ROOM
002036  101770                  BLOS    XMLOP1
002040  012721  002052  XMRTS:  MOV     #2$,(R1)+       ;SETUP RTS PC
002044  005021                  CLR     (R1)+
002046  005021                  CLR     (R1)+
002050  000207                  RTS     PC

002052  151456  RCTVAL: 151456
                        .DSABL  AMA

                        ;THIS IS THE BACKGROUND TASK WHICH WILL BE LOADED THRU
                        ; THE REMAINDER OF MEMORY

002054  000277  MEMTST: SCC                     ;SET CARRY BIT
002056  012727  123456          MOV     #123456,(PC)+   ;MEMDAT CONTAINS
002062  123456  MEMDAT: 123456
002064  106067  177773          RORB    MEMDAT+1        ;ROTATE LEFT BYTE OF MEMDAT
002070  103401                  BCS     .+4
002072  000000                  HALT                    ;C BIT WAS NOT SET
002074  102001                  BVC     .+4
002076  000000                  HALT                    ;V BIT WAS SET
002100  022767  151456  177754  CMP     #151456,MEMDAT  ;CHECK HERE FOR CORRECT ROTATE
002106  001401                  BEQ     .+4
002110  000000                  HALT                    ;ROTATE FAILED
002112  026737  177744  002052  CMP     MEMDAT,@#ROTVAL ;CHECK AGAIN REFERENCE LOW MEMORY
002120  001401                  BEQ     .+4
002122  000000                  HALT                    ;REF. TO LOW MEMORY FAILED
002124  000000  MEMEND: 0
002126  000000                  0
                        .ENABL  AMA
```

```
                                        ;INTERRUPT SERVICE FOR THE GT STOP INTERRUPT
                                        ;
002130  005777  177014        GTSTOP: TST     @GTSR           ;TEST STOP
002134  100403                        BMI     1$
002136  000000                        HALT                    ;ERROR, STOP INTERRUPT BUT NO STOP FLAG
002140  000137  001022                JMP     STARTB          ;RESTART TEST
                                        ;
002144  005337  001314        1$:     DEC     GTDLYD          ;DECREMENT DELAY
002150  001014                        BNE     GTST1           ;BRANCH IF NOT
002152  012737  000040  001314        MOV     #40,GTDLYD      ;RESET DELAY
002160  005237  001074                INC     GRPINC          ;UPDATE GRAPH INCREMENT
002164  022737  174110  003074        CMP     #STATSB!INCR+10.,GRPINC  ;TEST FOR INCREMENT
002172  001003                        BNE     GTST1           ;BRANCH IF NOT
002174  012737  174100  003074        MOV     #STATSB!INCR,GRPINC      ;RESET GRAPH INCREMENT
002202  012737  005752  005722  GTST1: MOV     #FILEOC,FILEOA
002210  012777  000001  176730        MOV     #1,@GTPC        ;RESUME THE DISPLAY
002216  000002                        RTI                     ;EXIT
                                        ;
002220  012737  005724  005722  GTLPEN: MOV    #FILEOB,FILEOA
002226  012777  000001  176712        MOV     #1,@GTPC        ;RESUME THE DISPLAY
002234  000002                        RTI
                                        ;
002236  000000                GTSHIF: HALT                    ;GT-40 SHIFT-OUT/TIME-OUT ERROR
002240  000137  001022                JMP     STARTB
```

```
                                        ;INTERRUPT SERVICE FOR THE DL PRINTER

002244  105777  176712        DLOUT:  TSTB    @DLOCSR         ;TEST FOR DONE
002250  100403                        BMI     .+10
002252  000000                        HALT                    ;ERROR, PRINTER INTERRUPT BUT NO PRINTER FLAG
002254  000137  001022                JMP     STARTB          ;RESTART TEST

002260  010446                        MOV     R4,-(SP)

002262  013704  002430        DLOUTA: MOV     PPNT,R4                 ;LOAD R4 WITH BYTE POINTER
002266  116437  005422  002434        MOVB    BUFF1(R4),PUNCHR        ;LOAD A CHARACTER TO BE OUTPUTTED
002274  005237  002430                INC     PPNT                    ;UPDATE CHARACTER POINTER
002300  022737  000050  002430        CMP     #40.,PPNT               ;TEST FOR END
002306  001002                        BNE     DLOUTB
002310  005037  002430                CLR     PPNT                    ;CLEAR PUNCH POINTER

002314  113777  002434  176642  DLOUTB: MOVB    PUNCHR,@DLODBR          ;OUTPUT A CHARACTER
002320  012604                        MOV     (SP)+,R4                ;RESTORE R4
002322  000002                        RTI                             ;EXIT

                                        ;INTERRUPT SERVICE FOR THE DL READER

002326  105777  176624        DLIN:   TSTB    @DLICSR                 ;TEST FOR DONE
002332  100403                        BMI     .+10
002334  000000                        HALT                            ;NOT DL INPUT FLAG
002336  000137  001022                JMP     STARTB                  ;RESTART TEST

002342  010446                        MOV     R4,-(SP)                ;SAVE R4
002344  013704  002432                MOV     RPNT,R4
002350  117737  176604  002436        MOVB    @DLIDBR,REDCHR          ;READ A CHARACTER
002356  042737  177600  002436        BIC     #177600,REDCHR          ;MASK CHARACTER
002364  113764  002436  005516        MOVB    REDCHR,BUFF2(R4)        ;PUT CHARACTER INTO THE BUFFER
002372  005237  002432                INC     RPNT                    ;UPDATE READ POINTER
002376  022737  000050  002432        CMP     #40.,RPNT               ;TEST FOR END
002404  001002                        BNE     DLINB
002406  005037  002432                CLR     RPNT
002412  013704  002432        DLINB:  MOV     RPNT,R4
002416  112764  000177  005516        MOVB    #177,BUFF2(R4)          ;ADD CURSOR
002424  012604                        MOV     (SP)+,R4                ;RESTORE R4
002426  000002                        RTI                             ;EXIT

002430  000000        PPNT:   0
002432  000000        RPNT:   0
002434  000240        PUNCHR: 240
002436  000240        REDCHR: 240
```

```
                                        ;INTERRUPT SERVICE FOR THE KEYBOARD

002440  105777  176546        KBIN:   TSTB    @TKS            ;TEST FOR DONE
002444  100403                        BMI     .+10
002446  000000                        HALT                    ;NOT KRB INPUT FLAG
002450  000137  001022                JMP     STARTB          ;RESTART

002454  010346                        MOV     R3,-(SP)        ;SAVE R3
002456  010446                        MOV     R4,-(SP)        ;SAVE R4
002460  013704  002624                MOV     KPNT,R4
002464  116737  176524  002626        MOVB    @TKB,KBCHR      ;READ CHARACTER
002472  042737  177600  002626        BIC     #177600,KBCHR   ;MASK
002500  113764  002626  005612        MOVB    KBCHR,BUFF3(4)  ;SAVE THE CHAR
002506  005237  002624                INC     KPNT            ;UPDATE POINTER
002512  022737  000050  002624        CMP     #40.,KPNT       ;TEST FOR END
002520  001002                        BNE     1$
002522  005037  002624                CLR     KPNT            ;CLEAR POINTER
002526  013704  002624        1$:     MOV     KPNT,R4
002532  116764  000177  005612        MOVB    #177,BUFF3(R4)  ;ADD CURSOR

                                        ;UPDATE OCTAL READOUT

002540  013703  002626                MOV     KBCHR,R3        ;GET CHAR
002544  004737  002610                JSR     PC,10$          ;LOAD BITS
002550  110437  005720                MOVB    R4,OCTA+2       ;SAVE BITS
002554  004737  002602                JSR     PC,11$          ;MOVE BITS
002560  110437  005700                MOVB    R4,OCTA+1       ;SAVE BITS
002564  004737  002602                JSR     PC,11$          ;MOVE BITS
002570  110437  005677                MOVB    R4,OCTA         ;SAVE BITS
002574  012604                        MOV     (SP)+,R4        ;RESTORE R4
002576  012603                        MOV     (SP)+,R3        ;RESTORE R3
002600  000002                        RTI                     ;EXIT

002602  006003        11$:    ROR     R3
002604  006003                ROR     R3
002606  006003                ROR     R3
002610  010304        10$:    MOV     R3,R4           ;LOAD R4
002612  042704  177770        BIC     #177770,R4      ;MASK BITS
002616  062704  000060        ADD     #60,R4          ;MAKE A NUMBER
002622  000207                RTS     PC

002624  000000        KPNT:   0
002626  000240        KBCHR:  240
```

```
                                        ;PART 1 OF THE BACKGROUND TASK
                                        ;
002630  012737  001000  003010  OVER:   MOV     #1000,PCOUNT    ;SET UP EXECUTION COUNT
                                        ;
                                        ;       COMPARE THE ROM DATA TO THE IMAGE DATA
                                        ;               R0=WORD NUMBER
                                        ;               R1=GOOD DATA
                                        ;               R2=GOOD DATA
                                        ;               R3=BAD ADDRESS
                                        ;               R4=BAD DATA
                                        ;
002636  012700  000000          BACK:   MOV     #0,%0           ;SETUP INITIAL WORD NUMBER
002642  013701  001020                  MOV     IMAGE,%1        ;SET UP BUFFER
002646  001415                          BEQ     TMEM            ;NO ROM SELECTED
002650  013703  001014                  MOV     ROMADD,%3       ;SET UP ROM ADDRESS
002654  011102          BACK1:  MOV     (%1),%2         ;READ A IMAGE WORD
002656  011304                  MOV     (%3),%4         ;READ A ROM WORD
002660  020204                  CMP     %2,%4           ;TEST FOR EQUAL
002662  001402                  BEQ     BACK2           ;BRANCH IF OK
002664  000000                  HALT                    ;ERROR ROM VALUE FAILED TO EQUAL THE
002666  000772                  BR      BACK1           ; THE EXPECTED
                                        ;
002670  022123          BACK2:  CMP     (%1)+,(%3)+     ;BUMP BOTH REGISTERS
002672  005200                  INC     %0              ;UPDATE WORD COUNTER
002674  023700  001016                  CMP     WORDS,%0        ;TEST FOR LAST WORD
002700  001365                  BNE     BACK1           ;BRANCK IF NOT LAST
                                        ;
                                        ;PART 2 OF THE BACKGROUND TASK
                                        ; EXECUTE WORSE CASE NOISE TEST THRU MEMORY
                                        ;
002702  004737  017000          TMEM:   JSR     PC,BUFFER       ;EXECUTE NOISE TEST
002706  005337  003010                  DEC     PCOUNT          ;DONE PASS ?
002712  001351                          BNE     BACK            ;NO
002714  012777  000001  176226          MOV     #1,@GTSR        ;YES RING THE BELL
002722  012777  000207  176270          MOV     #207,@TPB       ;RING THE BELL
002730  105777  176262          1$:     TSTB    @TPS
002734  100375                          BPL     1$
002736  012777  000207  176254          MOV     #207,@TPB
002744  105777  176246          2$:     TSTB    @TPS
002750  100375                          BPL     2$
002752  005737  000042                  TST     @#42            ;TEST LOC. 42
002756  001724                          BEQ     OVER            ;BR IF =0
002760  000005                          RESET
002762  000005                          RESET
002764  000005                          RESET
002766  013700  000042                  MOV     @#42,R0         ;READ VALUE
002772  004710          LOGICAL:        JSR     PC,(0)
002774  000240                  NOP
002776  000240                  NOP
003000  000240                  NOP
003002  000240                  NOP
003004  000137  001022                  JMP     STARTB
                                        ;
                                        ;
003010  000000          PCOUNT: 0
```

```
                                ;LINE THE EDGES OF THE SCREEN

003022  113004                          LONGV!INT4!LINE0                ;TOP LINE
003024  041777                          INTX!MAXX
003026  000000                          0
003030  110005                          LONGV!LINE1             ;RIGHT LINE
003032  040000                          INTX
003034  021377                          MINUSX!MAXY
003036  110006                          LONGV!LINE2             ;BOTTOM LINE
003040  061777                          INTX!MINUSX!MAXX
003042  000000                          0
003044  110007                          LONGV!LINE3             ;LEFT LINE
003046  040000                          INTX
003050  001377                          MAXY

                                ;SETUP THE X SINEWAVE

003052  114004                          POINT!LINE0
003054  000400                          400
003056  000200                          200
003060  110000                          LONGV
003062  041200                          INTX+1200                       ;DRAW X AXIS
003064  000000                          0
003066  114000                          POINT
003070  000440                          440
003072  000200                          200
003074  174104                  GRPINC: STATSB!INCR+4           ;GRAPHPLOT THE X SINEWAVE

003076  124000                          GRAPHY
003100  000200                           0200
003102  000205                           0205
003104  000212                           0212
003106  000217                           0217
003110  000224                           0224
003112  000231                           0231
003114  000236                           0236
003116  000243                           0243
003120  000247                           0247
003122  000253                           0253
003124  000257                           0257
003126  000262                           0262
003130  000265                           0265
003132  000270                           0270
003134  000272                           0272
003136  000274                           0274
003140  000276                           0276
003142  000277                           0277
003144  000277                           0277
```

```
  896  003146  000277                    0277
  897  003150  000277                    0277
  898  003152  000276                    0276
  899  003154  000275                    0275
  900  003156  000274                    0274
  901  003160  000272                    0272
  902  003162  000267                    0267
  903  003164  000264                    0264
  904  003166  000261                    0261
  905  003170  000256                    0256
  906  003172  000252                    0252
  907  003174  000246                    0246
  908  003176  000241                    0241
  909  003200  000235                    0235
  910  003202  000230                    0230
  911  003204  000223                    0223
  912  003206  000216                    0216
  913  003210  000211                    0211
  914  003212  000203                    0203
  915  003214  000176                    0176
  916  003216  000171                    0171
  917  003220  000163                    0163
  918  003222  000156                    0156
  919  003224  000151                    0151
  920  003226  000144                    0144
  921  003230  000137                    0137
  922  003232  000133                    0133
  923  003234  000127                    0127
  924  003236  000123                    0123
  925  003240  000117                    0117
  926  003242  000114                    0114
  927  003244  000111                    0111
  928  003246  000106                    0106
  929  003250  000104                    0104
  930  003252  000102                    0102
  931  003254  000101                    0101
  932  003256  000100                    0100
  933  003260  000100                    0100
  934  003262  000100                    0100
  935  003264  000100                    0100
  936  003266  000101                    0101
  937  003270  000102                    0102
  938  003272  000104                    0104
  939  003274  000106                    0106
  940  003276  000111                    0111
  941  003300  000113                    0113
  942  003302  000117                    0117
  943  003304  000122                    0122
  944  003306  000126                    0126
  945  003310  000132                    0132
  946  003312  000137                    0137
  947  003314  000144                    0144
  948  003316  000151                    0151
  949  003320  000156                    0156
  950  003322  000163                    0163
  951  003324  000170                    0170
```

952	003326	000175	0175
953	003330	000203	0203
954	003332	000210	0210
955	003334	000215	0215
956	003336	000222	0222
957	003340	000227	0227
958	003342	000234	0234
959	003344	000241	0241
960	003346	000245	0245
961	003350	000252	0252
962	003352	000255	0255
963	003354	000261	0261
964	003356	000264	0264
965	003360	000267	0267
966	003362	000271	0271
967	003364	000274	0274
968	003366	000275	0275
969	003370	000276	0276
970	003372	000277	0277
971	003374	000277	0277
972	003376	000277	0277
973	003400	000277	0277
974	003402	000276	0276
975	003404	000274	0274
976	003406	000273	0273
977	003410	000270	0270
978	003412	000266	0266
979	003414	000263	0263
980	003416	000257	0257
981	003420	000254	0254
982	003422	000247	0247
983	003424	000243	0243
984	003426	000237	0237
985	003430	000232	0232
986	003432	000225	0225
987	003434	000220	0220
988	003436	000213	0213
989	003440	000205	0205
990	003442	000200	0200
991	003444	000173	0173
992	003446	000165	0165
993	003450	000160	0160
994	003452	000153	0153
995	003454	000146	0146
996	003456	000141	0141
997	003460	000135	0135
998	003462	000130	0130
999	003464	000124	0124
1000	003466	000120	0120
1001	003470	000115	0115
1002	003472	000112	0112
1003	003474	000107	0107
1004	003476	000105	0105
1005	003500	000103	0103
1006	003502	000101	0101
1007	003504	000100	0100

```
1008  003506  000100                    0100
1009  003510  000100                    0100
1010  003512  000100                    0100
1011  003514  000100                    0100
1012  003516  000102                    0102
1013  003520  000103                    0103
1014  003522  000105                    0105
1015  003524  000107                    0107
1016  003526  000112                    0112
1017  003530  000115                    0115
1018  003532  000121                    0121
1019  003534  000125                    0125
1020  003536  000131                    0131
1021  003540  000135                    0135
1022  003542  000142                    0142
1023  003544  000147                    0147
1024  003546  000154                    0154
1025  003550  000161                    0161
1026  003552  000166                    0166
1027  003554  000173                    0173
1028
1029                            ;SETUP THE Y SINEWAVE
1030
1031  003556  114000                    POINT
1032  003560  000200                    200
1033  003562  000040                    40
1034  003564  110000                    LONGV                   ;DRAW Y AXIS
1035  003566  040000                    INTX
1036  003570  001200                    1200
1037  003572  114000                    POINT
1038  003574  000200                    200
1039  003576  000100                    100
1040  003600  120000                    GRAPHX          ;GRAPHPLOT THE Y SINEWAVE
1041
1042  003602  000200                     0200
1043  003604  000205                     0205
1044  003606  000212                     0212
1045  003610  000217                     0217
1046  003612  000224                     0224
1047  003614  000231                     0231
1048  003616  000236                     0236
1049  003620  000243                     0243
1050  003622  000247                     0247
1051  003624  000253                     0253
1052  003626  000257                     0257
1053  003630  000262                     0262
1054  003632  000265                     0265
1055  003634  000270                     0270
1056  003636  000272                     0272
1057  003640  000274                     0274
1058  003642  000276                     0276
1059  003644  000277                     0277
1060  003646  000277                     0277
1061  003650  000277                     0277
1062  003652  000277                     0277
1063  003654  000276                     0276
```

1064	003656	000275	0275
1065	003660	000274	0274
1066	003662	000272	0272
1067	003664	000267	0267
1068	003666	000264	0264
1069	003670	000261	0261
1070	003672	000256	0256
1071	003674	000252	0252
1072	003676	000246	0246
1073	003700	000241	0241
1074	003702	000235	0235
1075	003704	000230	0230
1076	003706	000223	0223
1077	003710	000216	0216
1078	003712	000211	0211
1079	003714	000203	0203
1080	003716	000176	0176
1081	003720	000171	0171
1082	003722	000163	0163
1083	003724	000156	0156
1084	003726	000151	0151
1085	003730	000144	0144
1086	003732	000137	0137
1087	003734	000133	0133
1088	003736	000127	0127
1089	003740	000123	0123
1090	003742	000117	0117
1091	003744	000114	0114
1092	003746	000111	0111
1093	003750	000106	0106
1094	003752	000104	0104
1095	003754	000102	0102
1096	003756	000101	0101
1097	003760	000100	0100
1098	003762	000100	0100
1099	003764	000100	0100
1100	003766	000100	0100
1101	003770	000101	0101
1102	003772	000102	0102
1103	003774	000104	0104
1104	003776	000106	0106
1105	004000	000111	0111
1106	004002	000113	0113
1107	004004	000117	0117
1108	004006	000122	0122
1109	004010	000126	0126
1110	004012	000132	0132
1111	004014	000137	0137
1112	004016	000144	0144
1113	004020	000151	0151
1114	004022	000156	0156
1115	004024	000163	0163
1116	004026	000170	0170
1117	004030	000175	0175
1118	004032	000203	0203
1119	004034	000210	0210

```
1120  004036  000215                    0215
1121  004040  000222                    0222
1122  004042  000227                    0227
1123  004044  000234                    0234
1124  004046  000241                    0241
1125  004050  000245                    0245
1126  004052  000252                    0252
1127  004054  000255                    0255
1128  004056  000261                    0261
1129  004060  000264                    0264
1130  004062  000267                    0267
1131  004064  000271                    0271
1132  004066  000274                    0274
1133  004070  000275                    0275
1134  004072  000276                    0276
1135  004074  000277                    0277
1136  004076  000277                    0277
1137  004100  000277                    0277
1138  004102  000277                    0277
1139  004104  000276                    0276
1140  004106  000274                    0274
1141  004110  000273                    0273
1142  004112  000270                    0270
1143  004114  000266                    0266
1144  004116  000263                    0263
1145  004120  000257                    0257
1146  004122  000254                    0254
1147  004124  000247                    0247
1148  004126  000243                    0243
1149  004130  000237                    0237
1150  004132  000232                    0232
1151  004134  000225                    0225
1152  004136  000220                    0220
1153  004140  000213                    0213
1154  004142  000205                    0205
1155  004144  000200                    0200
1156  004146  000173                    0173
1157  004150  000165                    0165
1158  004152  000160                    0160
1159  004154  000153                    0153
1160  004156  000146                    0146
1161  004160  000141                    0141
1162  004162  000135                    0135
1163  004164  000130                    0130
1164  004166  000124                    0124
1165  004170  000120                    0120
1166  004172  000115                    0115
1167  004174  000112                    0112
1168  004176  000107                    0107
1169  004200  000105                    0105
1170  004202  000103                    0103
1171  004204  000101                    0101
1172  004206  000100                    0100
1173  004210  000100                    0100
1174  004212  000100                    0100
1175  004214  000100                    0100
```

```
1176  004216  000100                    0100
1177  004220  000102                    0102
1178  004222  000103                    0103
1179  004224  000105                    0105
1180  004226  000107                    0107
1181  004230  000112                    0112
1182  004232  000115                    0115
1183  004234  000121                    0121
1184  004236  000125                    0125
1185  004240  000131                    0131
1186  004242  000135                    0135
1187  004244  000142                    0142
1188  004246  000147                    0147
1189  004250  000154                    0154
1190  004252  000161                    0161
1191  004254  000166                    0166
1192  004256  000173                    0173
1193
```

```
1194
1195                                    ;SETUP TO DISPLAY THE OCTAGONS
1196
1197  004260  114000                            POINT
1198  004262  001434                            1434
1199  004264  000724                            724
1200  004266  130030                            RELATV!BLKON
1201  004270  041600                            INTX+1600
1202  004272  041607                            INTX+1600+7
1203  004274  040007                            INTX+7
1204  004276  061607                            INTX!MINUSX+1600+7
1205  004300  061600                            INTX!MINUSX+1600
1206  004302  061707                            INTX!MINUSX+1600+MINSUY+7
1207  004304  040107                            INTX+MINSUY+7
1208  004306  041707                            INTX+1600+MINSUY+7
1209  004310  114000                            POINT
1210  004312  001430                            1430
1211  004314  000710                            710
1212  004316  130020                            RELATV!BLKOFF
1213  004320  043600                            INTX+3600
1214  004322  043617                            INTX+3600+17
1215  004324  040017                            INTX+17
1216  004326  063617                            INTX!MINUSX+3600+17
1217  004330  063600                            INTX!MINUSX+3600
1218  004332  063717                            INTX!MINUSX+3600+MINSUY+17
1219  004334  040117                            INTX+MINSUY+17
1220  004336  043717                            INTX+3600+MINSUY+17
1221  004340  114000                            POINT
1222  004342  001420                            1420
1223  004344  000660                            660
1224  004346  104030                            SHORTV!BLKON
1225  004350  047600                            INTX+7600
1226  004352  047637                            INTX+7600+37
1227  004354  040037                            INTX+37
1228  004356  067637                            INTX!MINUSX+7600+37
1229  004360  067600                            INTX!MINUSX+7600
1230  004362  067737                            INTX!MINUSX+7600+MINSUY+37
1231  004364  040137                            INTX+MINSUY+37
1232  004366  047737                            INTX+7600+MINSUY+37
1233  004370  114000                            POINT
1234  004372  001400                            1400
1235  004374  000600                            600
1236  004376  104020                            SHORTV!BLKOFF
1237  004400  057600                            INTX+17600
1238  004402  057677                            INTX+17600+77
1239  004404  040077                            INTX+77
1240  004406  077677                            INTX!MINUSX+17600+77
1241  004410  077600                            INTX!MINUSX+17600
1242  004412  077777                            INTX!MINUSX+17600+MINSUY+77
1243  004414  040177                            INTX+MINSUY+77
1244  004416  057777                            INTX+17600+MINSUY+77
1245  004420  114030                            POINT!BLKON
1246  004422  001360                            1360
1247  004424  000520                            520
1248  004426  110000                            LONGV                   ;OCTOGON BY LENGTH OF 137
1249  004430  040137                            INTX+137
```

```
 1250  004432  000000                    0
 1251  004434  040137                    INTX+137
 1252  004436  000137                    137
 1253  004440  040000                    INTX
 1254  004442  000137                    137
 1255  004444  060137                    INTX!MINUSX+137
 1256  004446  000137                    137
 1257  004450  060137                    INTX!MINUSX+137
 1258  004452  000000                    0
 1259  004454  060137                    INTX!MINUSX+137
 1260  004456  020137                    MINUSX+137
 1261  004460  040000                    INTX
 1262  004462  020137                    MINUSX+137
 1263  004464  040137                    INTX+137
 1264  004466  020137                    MINUSX+137
 1265  004470  114120                    POINT!BLKOFF!LPOFF
 1266  004472  001340                    1340
 1267  004474  000440                    440
 1268  004476  110000                    LONGV                          ;OCTOGON BY LENGTH OF 177
 1269  004500  040177                    INTX+177
 1270  004502  000000                    0
 1271  004504  040177                    INTX+177
 1272  004506  000177                    177
 1273  004510  040000                    INTX
 1274  004512  000177                    177
 1275  004514  060177                    INTX!MINUSX+177
 1276  004516  000177                    177
 1277  004520  060177                    INTX!MINUSX+177
 1278  004522  000000                    0
 1279  004524  060177                    INTX!MINUSX+177
 1280  004526  020177                    MINUSX+177
 1281  004530  040000                    INTX
 1282  004532  020177                    MINUSX+177
 1283  004534  040177                    INTX+177
 1284  004536  020177                    MINUSX+177
 1285
 1286  004540  114140                    POINT!LPON
 1287  004542  000144                    100
 1288  004544  001277                    MAXY-100
 1289  004546  164000                    DNOP
 1290  004550  170240                    STATSA!ITALO
 1291  004552  100000                    CHAR
 1292  004554  040500  04:502  042504    .ASCII  '@ABCDEFGHIJKLMNOPQRSTUVWXYZ'
 1293  004556  043506  044510  045512
 1294  004570  046514  047516  050520
 1295  004576  051522  052524  053526
 1296  004604  054530     132
 1297  004637     040  02:041  022043    .ASCIZ  @ !"#$%&'()*+,-./0123456789:;<=>?@
 1298  004614  023045  024047  025051
 1299  004622  026053  027055  030057
 1300  004630  031061  032063  033065
 1301  004636  034067  035071  036053
 1302  004644  037075  000077
 1303
 1304  004650  164000                    .EVEN
 1305  004652  170060                    DNOP
                                         STATSA!ITAL1
```

```
1336  004654  114000                          PCINT
1337  004656  000100                          100
1338  004660  001247                          MAXY-130
1339  004662  100000                          CHAR
1340  004664  040500  041502  042504          .ASCII  '@ABCDEFGHIJKLMNOPQRSTUVWXYZ'
1341  004672  043506  044510  045512
1342  004700  045514  047516  050520
1343  004706  051522  052524  053526
1344  004714  054530      132
1345  004717      040  021041  022043          .ASCIZ  @ !"#$%&'()*+,-./0123456789:;<=>?@
1346  004724  023045  024047  025051
1347  004730  026053  027055  030057
1348  004736  031061  032063  033065
1349  004744  034067  035071  036073
1350  004752  037075  000077
1351
1352  004756  170040                          .EVEN
1353  004760  114000                          STATSA!ITALO
1354  004762  000220                          POINT
1355  004764  001177                          220
1356  004766  100000                          MAXY-200
1357  004770                                  CHAR
1358  004772      140      141      142       .BYTE   140,141,142,143,144,145,146,147
1359  004775      143      144      145
1360  005000      146      147
1361  005002      150      151      152       .BYTE   150,151,152,153,154,155,156,157
1362  005005      153      154      155
1363  005010      156      157
1364  005012      160      161      162       .BYTE   160,161,162,163,164,165,166,167
1365  005015      163      164      165
1366  005020      166      167
1367  005022      170      171      172       .BYTE   170,171,172,173,174,175,176,177
1368  005025      173      174      175
1369  005030      176      177
1370
1371  005032  170060                          .EVEN
1372  005034  114000                          STATSA!ITALI
1373  005036  000220                          POINT
1374  005040  001147                          220
1375  005042  100000                          MAXY-230
1376  005044                                  CHAR
1377  005047      140      141      142       .BYTE   140,141,142,143,144,145,146,147
1378  005052      143      144      145
1379  005054      146      147
1380  005056      150      151      152       .BYTE   150,151,152,153,154,155,156,157
1381  005061      153      154      155
1382  005064      156      157
1383  005066      160      161      162       .BYTE   160,161,162,163,164,165,166,167
1384  005071      163      164      165
1385  005074      166      167
1386  005076      170      171      172       .BYTE   170,171,172,173,174,175,176,177
1387  005103      173      174      175
1388  005106      176      177
1389
1390  005110  170040                          .EVEN
1391  005112  114000                          STATSA!ITALO
1392  005114  000220                          POINT
1393  005116  001077                          220
                                              MAXY-300
```

```
1362  005114  100000                          CHAR
1363  005116     016     000     001          .BYTE   16,0,1,2,3,4,5,6,7,10,11,12,13,14,15,16
1364  005121     002     003     004
1365  005124     005     006     007
1366  005127     010     011     012
1367  005132     013     014     015
1368  005135     016
1369  005136     020     021     022          .BYTE   20,21,22,23,24,25,26,27,30,31,32,33,34,35,36,37,17,0
1370  005141     023     024     025
1371  005144     026     027     030
1372  005147     031     032     033
1373  005152     034     035     036
1374  005155     037     017     000
1375                                          .EVEN
1376  005160  170060                          STATSA!ITAL1
1377  005162  114000                          POINT
1378  005164  000220                          220
1379  005166  001047                          MAXY-330
1380  005170  100000                          CHAR
1381  005172     016     000     001          .BYTE   16,0,1,2,3,4,5,6,7,10,11,12,13,14,15,16
1382  005175     002     003     004
1383  005200     005     006     007
1384  005203     010     011     012
1385  005206     013     014     015
1386  005211     016
1387  005212     020     021     022          .BYTE   20,21,22,23,24,25,26,27,30,31,32,33,34,35,36,37,17,0
1388  005215     023     024     025
1389  005220     026     027     030
1390  005223     031     032     033
1391  005226     034     035     036
1392  005231     037     017     000
1393                                          .EVEN
1394  005234  170040                          STATSA!ITAL0
1395
1396                                    ;SETUP INTENSITY LEVEL TEST
1397  005236  114000                          POINT
1398  005240  000340                          340
1399  005242  001000                          1000
1400  005244  113604                          LONGV!INT7!LINE0
1401  005246  040400                          INTX+400
1402  005250  000000                          0
1403  005252  114000                          POINT
1404  005254  000340                          340
1405  005256  000740                          740
1406  005260  113400                          LONGV!INT6
1407  005262  040400                          INTX+400
1408  005264  000000                          0
1409  005266  114000                          POINT
1410  005270  000340                          340
1411  005272  000700                          700
1412  005274  113200                          LONGV!INT5
1413  005276  040400                          INTX+400
1414  005300  000000                          0
1415  005302  114000                          POINT
1416  005304  000340                          340
1417  005306  000640                          640
```

```
1418  005310  113000                  LONGV!INT4
1419  005312  040400                  INTX+400
1420  005314  000000                  0
1421  005316  114000                  POINT
1422  005320  000340                  340
1423  005322  000600                  600
1424  005324  112600                  LONGV!INT3
1425  005326  040400                  INTX+400
1426  005330  000000                  0
1427  005332  114000                  POINT
1428  005334  000340                  340
1429  005336  000540                  540
1430  005340  112400                  LONGV!INT2
1431  005342  040400                  INTX+400
1432  005344  000000                  0
1433  005346  114000                  POINT
1434  005350  000340                  340
1435  005352  000500                  500
1436  005354  112200                  LONGV!INT1
1437  005356  040400                  INTX+400
1438  005360  000000                  0
1439  005362  114000                  POINT
1440  005364  000340                  340
1441  005366  000440                  440
1442  005370  112000                  LONGV!INT0
1443  005372  040400                  INTX+400
1444  005374  000000                  0
1445
1446                          :SETUP THE MESSAGA BUFFERS
1447
1448  005376  117000                  POINT!INT4
1449  005400  000400                  400
1450  005402  000020                  20
1451  005404  100000                  CHAR
1452  005406  047503  020115  052517  .ASCII  /COM OUTPUT  /
1453  005414  050124  052125  020040
1454  005422  042504  043503  040522  BUFF1:  .ASCII  /DECGRAPHIC-11 DISPLAY TERMINAL GT4D VR14/
1455  005430  044120  041511  030455
1456  005436  020061  044504  050123
1457  005444  040514  020131  042524
1458  005452  046522  047111  046101
1459  005460  043440  032124  020060
1460  005466  051126  032061
1461  005472  114000                  POINT
1462  005474  000400                  400
1463  005476  000320                  320
1464  005500  100000                  CHAR
1465  005502  047503  027115  044440  .ASCII  /COM. INPUT  /
1466  005510  050116  052125  020040
1467  005516     000     000     000  BUFF2:  .BYTE   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1468  005521     000     000     000
1469  005524     000     000     000
1470  005527     000     000     000
1471  005532     000     000     000
1472  005535     000     000     000
1473  005540     000     000
```

```
1472  005542  000000  000000  000000          .BYTE   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1473  005545  000000  000000  000000
1474  005550  000000  000000  000000
1475  005553  000000  000000  000000
1476  005556  000000  000000  000000
1477  005561  000000  000000  000000
1478  005564  000000  000000
1479  005566  114000                          POINT
1480  005570  000400                          400
1481  005572  000350                          350
1482  005574  100000                          CHAR
1483  005576  051113  027102  044440          .ASCII  /KRB. INPUT  /
1484  006024  050116  052125  020040
1485  005612  000000  000000  000000  BUFF3:  .BYTE   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1486  005615  000000  000000  000000
1487  005620  000000  000000  000000
1488  005623  000000  000000  000000
1489  005626  000000  000000  000000
1490  005631  000000  000000  000000
1491  005634  000000  000000
1492  005636  000000  000000  000000          .BYTE   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1493  005641  000000  000000  000000
1494  005644  000000  000000  000000
1495  005647  000000  000000  000000
1496  005652  000000  000000  000000
1497  005655  000000  000000  000000
1498  005660  000000  000000
1499
1500  005662  114000                          POINT
1501  005664  000400                          400
1502  005666  000400                          400
1503  005670  100000                          CHAR
1504  005672  041517  020124  000000          .ASCIZ  /OCT /
1505  005677  000000  000000  000000  OCTA:   .BYTE   0,0,0
1506  005702  164000                          DNOP
1507  005704  164000                          DNOP
1508  005706  164000                          DNOP
1509  005710  164000                          DNOP
1510  005712  164000                          DNOP
1511  005714  164000                          DNOP
1512  005716  164000                          DNOP
1513  005720  160000                          DJMP
1514  005722  005752                  FILEOA: FILEOC
1515  005724  114000                  FILEOB: POINT
1516  005726  001000                          1000
1517  005730  000440                          440
1518  005732  100000                          CHAR
1519  005734  044514  044107  026524          .ASCIZ  /LIGHT-PEN HIT/
1520  005742  042520  020116  044510
1521  005750  000124
1522
1523  005752  173400                  FILEOC: DSTOP
1524  005754  160000                          DJMP
1525  005756  003012                          FILEOO
1526
```

```
                        ;*********************************************************
                        ;       EXCEPT FOR THE NEW ORGIN ADDRESS AND SEVERAL "!160000"
                        ;       FOR ADDRESS FUDGING THIS IS AN EXACT COPY OF THE CONTENTS
                        ;       OF THE GT-40 BOOTSTRAP VERSION #2
                        ;*********************************************************


                                .TITLE  SCROLLING ROM BOOTSTRAP FOR THE GT40



                        ;                       BOOTGT.T16      OCT 10, 1973



                        ;               COPYRIGHT 1973, DIGITAL EQUIPMENT CORPORATION
                        ;               146 MAIN STREET
                        ;               MAYNARD, MASSACHUSETTS
                        ;                                       01754



                        ;               WRITTEN BY JACK BURNESS.



                        ;               THIS PROGRAM IS THE SECOND VERSION THE THE ROM BOOTSTRAP FOR
                        ;       THE GT40 DISPLAY TERMINAL. IT INCLUDES SCROLLING AND AN END OF
                        ;       MEMORY SEARCH FOR THE LOADER.


                                .ENABL  ABS,AMA                 ;ASSEMBLER DIRECTIVES FOR ABSOLUTE BINARY OUTPUT
                                                                ;  NOTE: USE "MACDLX" TO ASSEMBLE THIS PROGRAM.



                                .SBTTL  DEFINITION SECTION
```

```
1582
1583
1584                            ;                    REGISTER DEFINITIONS
1585                            ;                    -------- -----------
1586
1587
1588
1589
1590                            ;                BASIC DEFINTIONS
1591                            ;                ----- ----------
1592
1593            000000                  R0=%0                           ;DEFINE STANDARD VALJES.
1594            000001                  R1=%1
1595            000002                  R2=%2
1596            000003                  R3=%3
1597            000004                  R4=%4
1598            000005                  R5=%5
1599            000006                  SP=%6
1600            000007                  PC=%7
1601
1602
1603
1604                            ;                GT40 DEFINTIONS
1605                            ;                ---- ----------
1606
1607
1608            000000                  CHAR=R0                         ;CONTAINS THE INPUT CHARACTER.
1609            000001                  POINTR=R1                       ;POINTS TO NEXT INSEPTION BYTE IN DISPLAY BUFFER
1610            000002                  TABCNT=R2                       ;CHARACTER COUNTER FOR THE "TAB" FEATURE.
1611            000003                  SCAN=R3                         ;GENERALLY CONTAINS A POINTER WHICH
1612                                                                    ;IS USED WHEN SCANNING MEMORY FOR SOMETHING.
1613            000004                  HOLD=R4                         ;TYPICALLY A TEMPORARY WHICH IS USED TO RETAIN
1614                                                                    ;A VALUE FOR A SHORT TIME.
1615            000005                  COUNTR=R5                       ;TYPICALLY USED AS A COUNTER.
1616
1617
1618
1619
1620
1621                            ;                LOADER DEFINITIONS
1622                            ;                ------ ----------
1623
1624
1625
1626
1627
1628            000000                  L.BYT=CHAR                      ;CHARACTER INPUT FOR THE LOADER.
1629            000001                  L.ADR=POINTR                    ;CURRENT MEMORY ADDRESS TO BE LOADED.
1630            000002                  L.BC=TABCNT                     ;NUMBER OF DATA ITEMS TO LOAD.
1631            000005                  L.CKSM=COUNTR                   ;CHECKSUM ON THE INPUT DATA.
1632            000003                  INDEX=SCAN                      ;INDICATES HOW TO ASSEMBLE THE 8 BIT CHARACTER.
1633
1634
```

```
1635
1636
1637                            ;                    MAJOR SYSTEM DEFINITIONS
1638                            ;                    ----- ------ -----------
1639
1640
1641
1642    166000                      ORIGIN=166000               ;ORIGIN OF THE BOOTSTRAP.
1643
1644    175610                      DL11IS=175610               ;INPUT STATUS REGISTER OF DL11
1645    175612                      DL11IB=DL11IS+2             ;INPUT CHARACTER FROM DL11
1646    175614                      DL11OS=DL11IB+2             ;OUTPUT STATUS OF THE DL11
1647    175616                      DL11OB=DL11OS+2             ;OUTPUT CHARACTER TO THE DL11
1648
1649    177560                      KBDIS=177560                ;KEYBOARD INPUT STATUS
1650    177562                      KBDIB=KBDIS+2               ;CURRENT CHARACTER FROM KEYBOARD.
1651
1652    172000                      GT40PC=172000               ;GT40 PROGRAM COUNTER.
1653    172002                      GT40SR=GT40PC+2             ;GT40 STATUS REGISTER ADDRESS.
1654
1655
1656    001000                      BSTART=1000                 ;START OF THE DISPLAY BUFFER
1657    007000                      BLIMIT=7000                 ;APPROXIMATE END OF THE DISPLAY BUFFER.
1658    007776                      TMPEND=7776                 ;LOCATION OF INITIALIZATION STACK.
1659    000004                      CORSTR=4                    ;LOCATION OF PDP-11 TRAP VECTOR.
1660    007012                      JMPADD=BLIMIT+10.           ;WHERE THE POINTER IS TO FIRST CHAR ON SCREEN
1661    000040                      NUMLIN=32.                  ;NUMBER OF LINES ON TEXT TO SHOW ON THE SCREEN
1662
1663    005015                      CRLF=5015                   ;CARRIAGE RETURN - LINE FEED
1664    000175                      ALTMOD=175                  ;THE "KEY" CHARACTER [I.E. ALTMODE].
1665
1666    160000                      DISJMP=160000               ;THE GT40 JMP INSTRUCTION
1667    173000                      DISTOP=173000               ;THE GT40 STOP DISPLAY INSTRUCTION.
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686                            .SBTTL  INITIALIZATION AND RESTART CODE
```

```
                                ;               GT40 BOOTSTRAP CODE
                                ;               ---- --------- ----

                006000                  .=6000
                                ;       .=ORIGIN                        ;DEFINE ORIGIN OF THE BOOTSTRAP.

                                ;               COLD INITIALIZATION CODE
                                ;               ---- -------------- ----

006000  000005          START:  RESET                                   ;RESET ALL HARDWARE NOW.
006002  012737  000007  175610  MOV     #7,DL11IS                       ;INITIALIZE DL-11 INPUT NOW.
006010  012706  007776          MOV     #TMPEND,SP                      ;ESTABLISH A GOOD TEMPORARY STACK
                                                                        ;POINTER FOR CORE SEARCH.
006014  005237  175614          INC     DL11OS                          ;SET BREAK BIT
006020  004337  166652          JSR     SCAN,OUTLIT!160000              ;FOR 2 CHARACTER TIMES
006024  000000                  .WORD 0                                 ;SEND TWO ZERO'S

006026  012703  000004          MOV     #CORSTR,SCAN                    ;GET ADDRESS OF BAD CORE TRAP VECTOR.
006032  012723  166042          MOV     #NOTHERE!160000,(SCAN)+         ;AND INSERT A POINTER TO US THERE.

006036  005023          ENDCOR: CLR     (SCAN)+                         ;NOW CLEAR ALL OF MEMORY BEYOND THE POINTER,
006040  000776                  BR      ENDCOR                          ;UNTIL WE RUN OUT OF MEMORY AND TRAP.

006042  005743          NOTHER: TST     -(SCAN)                         ;WHEN WE TRAP OUT, WE COME HERE.
                                                                        ;WE BACK UP POINTER TO GOOD CORE.
                                                                        ;NOTE THAT IF WE TRAP OUT AGAIN, IT
                                                                        ;IS STILL OK, BECAUSE WE WILL LOOP
                                                                        ;UNTIL WE GET A GOOD CORE ADDRESS.
006044  010306                  MOV     SCAN,SP                         ;WHEN WE GET ONE, THAT IS LAST LOCATION
                                                                        ;IN THE MACHINE, AND HENCE OUR SP.
006046  105737  175614  1$:     TSTB    DL11OS                          ;SEE IF BREAK IS DONE
006052  100375                  BPL     1$                              ;NO GO BACK
006054  005037  175614          CLR     DL11OS                          ;CLEAR BREAK BIT

                                ;               RESTART INITIALIZATION CODE WHEN COMMUNICATIONS IS WORKING.
                                ;               ------- -------------- ---- ---- ------------- -- --------
```

```
1743
1744
1745
1746  006060  052706  007776      RESTRT: BIS     #TMPEND,SP              ;FORCE THE SP TO LIMIT OF EXISTING CORE.
1747
1748
1749  006064  012703  006700              MOV     #BLIMIT-NUMLIN-NUMLIN,SCAN      ;NOW WE WILL FILL THE KEY AREAS OF THE
1750  006070  012702  000040              MOV     #NUMLIN,TABCNT          ;DISPLAY BUFFER WITH INITIAL CR-LF'S.
1751
1752  006074  012723  005015      SETLP1: MOV     #CR_F,(SCAN)+           ;INSERT A CRLF NOW.
1753  006100  005302                      DEC     TABCNT                  ;AND LOOP UNTIL DONE.
1754  006102  003374                      BGT     SETLP1                  ;THUS DISPLAY CORE IS ALMOST CORRECT.
1755
1756
1757  006104  012703  166432              MOV     #SETUP!160000,SCAN              ;NOW WE WILL INITALIZE CORE FOR THE
1758                                                                      ;DISPLAY. PICK UP POINTER TO LIST.
1759
1760  006110  012302      SETLP2: MOV     (SCAN)+,TABCNT          ;GET NUMBER OF ITEMS TO INSERT.
1761  006112  001405              BEQ     SETDUN                  ;IF ZERO, WE ARE DONE.
1762  006114  012301              MOV     (SCAN)+,POINTR          ;PICK UP FIRST CORE ADDRESS POINTER.
1763
1764  006116  012321      SETLP3: MOV     (SCAN)+,(POINTR)+       ;MOVE OVER A DATA ITEM NOW.
1765  006120  005302              DEC     TABCNT                  ;ALL DONE?
1766  006122  003375              BGT     SETLP3                  ;NOPE. MOVE OVER THE NEXT.
1767  006124  000771              BR      SETLP2                  ;YES. GET NEXT MAJOR LIST TO INSERT.
1768
1769
1770  006126  012701  006776      SETDUN: MOV     #BLIMIT-2,POINTR        ;ESTABLISH THE BUFFER POINTER NOW.
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782                                  .SBTTL  VT05 SIMULATOR
```

```
                                ;               VT05 (SCROLLING) PORTION OF THE BOOTSTRAP
                                ;               ----  ---------  ------- -- --- ---------
                                
006132  004737  166564  NXTCHR: JSR     PC,GETCHR!160000C       ;GET A CHARACTER NOW.
006136  020027  000177          CMP     CHAR,#177               ;IS IT OUT OF RANGE?
006142  002373                  BGE     NXTCHR                  ;YEP. GET ANOTHER ONE.
006144  020027  000040          CMP     CHAR,#40                ;IS IT A PRINTING CHARACTER?
006150  002020                  BGE     NORMAL                  ;YES. IT'S A NORMAL PRINTING CHARACTER.
006152  010003                  MOV     CHAR,SCAN               ;MOVE IT OVER SO WE CAN PLAY WITH IT.
006154  162703  000007          SUB     #7,SCAN                 ;BIAS SO THAT BELL [7] IS ZERO.
006160  020327  000007          CMP     SCAN,#7                 ;IF CHARACTER IS LESS THEN BELL OR
006164  103362                  BHIS    NXTCHR                  ;GREATER THEN CR, THEN IGNORE.
006166  006303                  ASL     SCAN                    ;IF GOOD, MAKE IT WORD INDEX.
006170  060307                  ADD     SCAN,PC                 ;AND GO TO THE CORRECT ROUTINE.

006172  000426                  BR      BELL                    ;7=BELL
006174  000406                  BR      NORMAL                  ;10=BACKSPACE
006176  000411                  BR      TAB                     ;11=TAB
006200  000437                  BR      LF                      ;12=LINE FEED [LF]
006202  000420                  BR      VT                      ;13=VERTICAL TAB [VT]
006204  000424                  BR      FF                      ;14=FORM FEED [FF]
                                                                ;15=CARRIAGE RETURN [CR]


006206  012702  177777  CR:     MOV     #-1,TABCNT              ;RESET TAB POSITION ON A CR, AND
                                                                ;FALL THROUGH TO INSERT THE CHARACTER.

006212  004737  166350  NORMAL: JSR     PC,INSERT!160000        ;INSERT THE CHARACTER IN THE BUFFER.
006216  005202                  INC     TABCNT                  ;UPDATE TAB POSITION NOW.
006220  000744                  BR      NXTCHR                  ;AND GET NEXT CHARACTER.


006222  012700  000040  TAB:    MOV     #40,CHAR                ;ON A TAB, INSERT BLANKS UNTIL THE
006226  004737  166350          JSR     PC,INSERT!160000        ;NEXT CHARACTER POSITION IS A MULTIPLE
006232  005202                  INC     TABCNT                  ;OF 8.
006234  032702  000007          BIT     #7,TABCNT               ;ARE WE DONE YET?
006240  001370                  BNE     TAB                     ;NOPE.
006242  000733                  BR      NXTCHR                  ;YES.


006244  111705          VT:     MOVB    (PC),COUNTR             ;THIS PUTS THE LOW BYTE OF THE
                                                                ;BRANCH CODE IN COUNTR-SAVE A WORD
006246  000405                  BR      FFLOOP
                                                                ;
006250  005037  172002  BELL:   CLR     GT40SR                  ;RING BELL -WRITE IN GT40SR
006254  000726                  BR      NXTCHR                  ;AND LOOP BACK

006256  012705  000040  FF:     MOV     #NUMLIN,COUNTR          ;FORM FEED IS DONE BY INSERTING LF'S.
```

```
1839
1840  006262  012700  000012          FFLOOP: MOV     #12.CHAR                ;MAKE THE CHARACTER A LINEFEED.
1841  006266  004737  166304                  JSR     PC,LFSUB!160000         ;DO A LINEFEED.
1842  006272  005305                          DEC     COUNTR                  ;DONE?
1843  006274  003372                          BGT     FFLOOP                  ;NOPE. KEEP SENDING THEM.
1844  006276  000715                          BR      NXTCHR                  ;YES. NOW RETURN. DO NOT FALL THROUGH.
1845
1846
1847  006300  012746  166132          LF:     MOV     #NXTCHR!160000,-(SP)            ;RETURN TO NXTCHR AFTER PROCESSING
1848                                                                          ;THE LF BY FAKING A JSR.
1849
1850  006304  013703  007012          LFSUB:  MOV     JMPADD,SCAN             ;GET POINTER TO FIRST CHAR ON SCREEN
1851
1852  006310  122300                  LFLOOP: CMPB    (SCAN)+,CHAR            ;AND LOOK FOR A LINEFEED.
1853  006312  001406                          BEQ     LFOUND                  ;GOT IT. SEARCH HAS ENDED.
1854  006314  020327  007000                  CMP     SCAN,#BLIMIT            ;ARE WE AT END OF BUFFER?
1855  006320  103773                          BLO     LFLOOP                  ;NOPE. KEEP ON LOOKING.
1856  006322  012703  001000                  MOV     #BSTART,SCAN            ;IF AT TOP, RESET TO BOTTOM OF BUFFER
1857  006326  000770                          BR      LFLOOP                  ;AND KEEP ON LOOKING.
1858
1859  006330  005203                  LFOUND: INC     SCAN                    ;WE'VE GOT THE LINE FEED. STOP SHOWING
1860  006332  042703  000001                  BIC     #1,SCAN                 ;FIRST LINE BY CHANGING THE "DISJMP"
1861  006336  010337  007012                  MOV     SCAN,JMPADD             ;INSTRUCTION TO FIRST CHAR BEYOND LF.
1862  006342  004737  166350                  JSR     PC,INSERT!160000                ;INSERT THE LF IN THE BUFFER.
1863  006346  005000                          CLR     CHAR                    ;AND THEN INSERT ONE NULL CHARACTER BECAUSE
1864                                                                          ;THE "DISJMP" ADDRESS MUST BE EVEN, AND
1865                                                                          ;THIS GUARANTEES WE WILL NOT LOSE A
1866                                                                          ;A GOOD DATA CHARACTER. WE FALL THROUGH
1867                                                                          ;TO INSERT THE NULL IN THE BUFFER.
1868
1869
1870  006350  110021                  INSERT: MOVB    CHAR,(POINTR)+          ;STICK IN THE CHARACTER NOW.
1871  006352  032701  000001                  BIT     #1,POINTR               ;IS NEXT POSITION EVEN OR ODD?
1872  006356  001021                          BNE     INSRTX                  ;ODD. NO PROBLEMS. SPACE IS ALLOCATED.
1873  006360  020127  007000                  CMP     POINTR,#BLIMIT          ;EVEN. ARE WE AT THE END OF THE BUFFER?
1874  006364  103410                          BLO     INSRTL                  ;NO. JUST MAKE ROOM FOR ANOTHER WORD.
1875  006366  010103                          MOV     POINTR,SCAN             ;AT THE END. MOVE THE STUFF TO THE
1876  006370  012701  001000                  MOV     #BSTART,POINTR          ;BEGINNING OF THE BUFFER.
1877  006374  004737  166406                  JSR     PC,INSRTL!160000                ;CALL THE ROUTINE TO SAVE SPACE.
1878  006400  005023                          CLR     (SCAN)+                 ;AND CLEAR UP THE INSTRUCTIONS AT THE
1879  006402  005013                          CLR     (SCAN)                  ;END OF THE BUFFER.
1880  006404  000207                          RTS     PC                      ;AND THEN RETURN.
1881
1882  006406  022121                  INSRTL: CMP     (POINTR)+,(POINTR)+     ;BYPASS THE "DISJMP" BY ADDING 4 TO POINTR.
1883  006410  012711  166474                  MOV     #HEADER!160000,(POINTR) ;NOW INSERT THE DISJMP INSTRUCTION TO OUR HEADER
1884  006414  012741  160000                  MOV     #DISJMP,-(POINTR)       ;AND IT'S ADDRESS (PUT THEM IN BACKWARDS).
1885  006420  005041                          CLR     -(POINTR)               ;MAKE AVAILABLE A NEW CHARACTER SPOT.
1886
1887  006422  000207                  INSRTX: RTS     PC                      ;FINALLY RETURN TO THE CALLER.
1888
1889
1890
1891
1892
1893  006424  012737  001000  172000  GTBUSE: MOV     #BSTART,GT40PC          ;ON A BUS ERROR, WE MERELY RESTART THE GT40 AT
1894
```

```
1895                                                    ;THE RTI FOR THIS ROUTINE
1896                                                    ;IS THE FIRST WORD OF THE TABLE
1897                                                    ;BELOW-IT SAVES A WORD!
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909                            ;               INITIALIZATION TABLE FOR THE SCROLLER
1910                            ;               -------------- ----- --- --- --------
1911
1912
1913
1914  006432  000002    SETUP:  .WORD   2                ;INITIALIZE 2 WORDS.--ALSO RTI FROM ABOVE
1915  006434  000330            .WORD   330              ;STARTING AT LOCATION 330
1916  006436  166424            .WORD   GTBUSE!160000           ;FIRST WORD IS POINTER TO BUS ERROR ROUT
1917  006440  000200            .WORD   200              ;SECOND WORD IS NEW STATUS WORD ON INTERUPT.
1918
1919  006442  000007            .WORD   7                ;INITIALIZE THE END OF THE BUFFER TO
1920  006444  006776            .WORD   BLIMIT-2         ;A CLEAR SPACE TO INSERT THE CHARACTER.
1921  006446  000000            .WORD   0                ;THIS IS THE "RUNNING" START. THIS IS
1922  006450  160000  166474    .WORD   DISJMP,HEADER!160000            ;FOLLOWED BY A DISJMP TO OUR HEADER BLOC
1923  006454  160000  001000    .WORD   DISJMP,BSTART    ;AND THEN A DISJMP TO THE START OF THE BUFFER
1924  006460  160000  006700    .WORD   DISJMP,BLIMIT-NUMLIN-NUMLIN     ;AND A DISJMP TO THE FIRST CHAR ON SCREE
1925
1926  006464  000001            .WORD   1                ;FINALLY START THE GT40 GOING AT
1927  006466  172000            .WORD   GT40PC           ;THE POSITION INSTRUCTION IN THE
1928  006470  166474            .WORD   HEADER!160000           ;HEADER BLOCK.
1929
1930  006472  000000            .WORD   0                ;END OF INIT CODE
1931
1932                            ;               HEADER BLOCK FOR THE SCROLLER
1933
1934  006474  103334    HEADER: .WORD   103334           ;ENABL CHAR MODE,BLINKING
1935  006476  000177            .WORD   177              ;A BLINKING BOX-RUB OUT!
1936  006500  116124            .WORD   116124           ;GO TO POINT MODE
1937  006502  171340            .WORD   171340           ;LOAD STATUS REGISTER
1938  006504  000000  001352    .WORD   0,1352           ;POINT TO UPPER LEFT
1939  006510  103324            .WORD   103324           ;BACK TO CHAR MODE
1940  006512  160000  007010    .WORD   DISJMP,JMPADD-2  ;AND TO THE CHANGING JMP INST.
1941
1942                            .SBTTL  COMMUNICATIONS AND MISC. SUPPORT ROUTINES
```

```
                        ;                       COMMUNICATIONS HANDLING ROUTINES
                        ;                       ------------- -------- --------

                        ;
                        ;                       THE DL-11 HANDLER
                        ;                       --- ----- -------

006516  105737  175610        GETDL:  TSTB    DL11IS          ;CHECK THE HOST INPUT STATUS.
006522  100011                        BPL     GETDL1          ;HOST DID NOT SEND ANYTHING, YET.
006524  113700  175612                MOVB    DL11IB,CHAR     ;HOST SENT US A CHARACTER. PROCESS IT.
006530  012737  000007  175610        MOV     #7,DL11IS       ;REENABLE THE HOST TELECOMMUNICATIONS.
006536  042700  177600                BIC     #-200,CHAR      ;MAKE CHARACTER JUST SEVEN BITS.
006542  001765                        BEQ     GETDL           ;IF NULL, IGNORE IT.
006544  000207                        RTS     PC              ;ELSE RETURN NOW.

006546  105737  177560        GETDL1: TSTB    KBDIS           ;DID USER TYPE A CHARACTER?
006552  100361                        BPL     GETDL           ;NO. GO BACK AND CHECK HOST MACHINE.
006554  113737  177562  175616        MOVB    KBDIB,DL110B    ;MOVE THE CHARACTER TO THE HOST.
006562  000755                        BR      GETDL           ;AND CHECK AGAIN FOR INPUT.


                        ;                       THE "GET CHARACTER" ROUTINE
                        ;                       --- --- --------- -------

006564  004737  166516        GETCHR: JSR     PC,GETDL!160000 ;GET A CHARACTER FROM THE HOST NOW.
006570  020027  000175                CMP     CHAR,#ALTMOD    ;IS IT AN "ALTMODE"
006574  001025                        BNE     GETEXT          ;NO. EXIT NOW.

006576  004737  166516                JSR     PC,GETDL!160000 ;YES. GET ANOTHER ONE NOW.
006602  020027  000114                CMP     CHAR,#'L        ;IS IT AN "L"
006606  001501                        BEQ     LOADER          ;YES. START LOADING NOW.
006610  020027  000122                CMP     CHAR,#'R        ;IS IT AN "R"
006614  001015                        BNE     GETEXT          ;NO. IGNORE THE ALTMODE AND JUST RETURN THE CHAR

006616  012737  173030  007010        MOV     #DISTOP,JMPADD-2 ;YES. RESET. STOP DISPLAY BY INSERTING A "DISTOP
006624  000137  166060        PRESTR: JMP     RESTRT!160000   ;INSTRUCTION IN THE BUFFER, AND RESTART.


                        ;               THE "GET A SIX BIT CHARACTER" ROUTINE
```

```
                              ;         ---  --- - --- --- ---------  -------

 006630  004737  166564       GETSIX: JSR     PC,GETCHR!160000               ;GET A CHARACTER NOW.
 006634  020027  000040               CMP     CHAR,#40               ;IS IT A LEGAL PRINTING CHARACTER?
 006640  002517                       BLT     L.BAD                  ;NOPE. ABORT
 006642  020027  000137               CMP     CHAR,#137              ;IT'S BIG ENOUGH. IS IT TOO BIG?
 006646  003114                       BGT     L.BAD                  ;YEP. ABORT.

 006650  000207               GETEXT: RTS     PC                     ;RETURN TO THE CALLER.


                              ;         THIS OUTPUTS TWO CHARACTERS VIA A
                              ;         JSR SCAN,OUTLIT
                              ;         'TWO CHARACTERS'

 006652  112337  175616       OUTLIT: MOVB    (SCAN)+,DL110B
 006656  112337  175616               MOVB    (SCAN)+,DL110B         ;DOUBLE BUFFERED
 006662  000203                       RTS     SCAN                   ;RETURN







                              ;         THE "GET AN EIGHT BIT CHARACTER" ROUTINE
                              ;         ---  --- -- ----- --- ---------  -------



                              ;                 THIS ROUTINE DIFFERS FROM THE PREVIOUS ROUTINES
                              ;         IN THAT IT WILL TAKE SIX BIT CHARACTERS AND ASSEMBLE
                              ;         THEM FOR THE LOADER TO USE. NOTE THAT FROM THIS POINT
                              ;         ON WE WILL SWITCH TO THE LOADER DEFINITIONS OF THE
                              ;         REGISTERS. THUS THE CHARACTER IS RETURNED IN
                              ;         REGISTER "L.BYT" RATHER THAN CHAR (THOUGH THEY ARE
                              ;         PHYSICALLY THE SAME).


 006664  004737  166630       GET8:   JSR     PC,GETSIX!160000               ;GET A SIXBIT CHARACTER.
 006670  010046                       MOV     L.BYT,-(SP)            ;SAVE IT ON THE STACK.
 006672  005723                       TST     (INDEX)+               ;UPDATE INDEX TO NEXT ITEM (ALL ARE *2)
 006674  000163  166676               JMP     GET8TB-2!160000(INDEX)         ;AND DISPATCH ACCORDING TO THE INDEX.

 006700  000404               GET8TB: BR      GET81                  ;INDEX=2: ASSEMBLE FIRST CHAR
 006702  000416                       BR      GET82                  ;INDEX=4: ASSEMBLE SECOND CHAR
 006704  000432                       BR      GET83                  ;INDEX=6: ASSEMBLE THIRD AND LAST CHAR
                                                                     ;INDEX=8: RESET INDEX TO 0 [2] AND RETRY.


 006706  012703  000002       GET84:  MOV     #2,INDEX               ;THE FOURTH INDEX IS THE SAME AS THE FIRST
                                                                     ;INDEX. JUST RESET IT AND FALL THROUGH.
```

```
                                                                ;GET ANOTHER CHARACTER NOW.
006712  004737  166630  GET81:  JSR     PC,GETSIX!160000
006716  010004                  MOV     L.BYT,HOLD      ;AND PRESERVE IT FOR NEXT TIME THROUGH.
006720  006300                  ASL     L.BYT           ;NOW THROW AWAY LEFT MOST BITS OF
006722  006300                  ASL     L.BYT           ;THE 8 BIT CHARACTER. NOW MERGE IN
006724  106300                  ASLB    L.BYT           ;THE LEFT TWO BITS OF THE
006726  106116                  ROLB    (SP)            ;NEW SIX BIT CHARACTER WITH THE SIX
006730  106300                  ASLB    L.BYT           ;BITS FROM THE CHARACTER ON THE
006732  106116                  ROLB    (SP)            ;STACK. IST CHARACTER IS NOW ASSEMBLED.
006734  012600                  MOV     (SP)+,L.BYT     ;SO WE'LL RETURN IT TO THE USER.
006736  000207                  RTS     PC              ;AND THEN WE SHALL RETURN TO HIM.

006740  006300  GET82:  ASL     L.BYT           ;THE SECOND CHARACTER IS CREATED FROM
006742  006300                  ASL     L.BYT           ;THE 4 RIGHT BITS OF THE PREVIOUS CHARACTER
006744  106300                  ASLB    L.BYT           ;AND THE FOUR MIDDLE BITS OF THE PRESENT
006746  106104                  ROLB    HOLD            ;8 BIT CHARACTER.
006750  106300                  ASLB    L.BYT           ;WE WILL CREATE THE NEW 8 BIT
006752  106104                  ROLB    HOLD            ;IN THIS REGISTER, SINCE IT
006754  106300                  ASLB    L.BYT           ;MORE CONVIENT. WE WILL MOVE OVER THE
006756  106104                  ROLB    HOLD            ;ANSWER AT THE END.
006760  106300                  ASLB    L.BYT           ;ONE MORE TO GO
006762  106104                  ROLB    HOLD            ;DONE.
006764  010400                  MOV     HOLD,L.BYT      ;BRING OVER THE VALUE.
006766  012604                  MOV     (SP)+,HOLD      ;AND REMEMBER THE LAST CHARACTER WE RECEIVED.
006770  000207                  RTS     PC              ;AND RETURN TO THE CALLER.


006772  006100  GET83:  ROL     L.BYT           ;FINAL CHARACTER IS EASY. JUST A
006774  106100                  ROLB    L.BYT           ;SIMPLE MERGER OF LEFT TWO BITS OF
006776  006004                  ROR     HOLD            ;PREVIOUS VALUE WITH RIGHT SIX BITS
007000  106000                  RORB    L.BYT           ;OF LAST (4TH) CHARACTER RECEIVED.
007002  006004                  ROR     HOLD
007004  106000                  RORB    L.BYT           ;AND WE ARE DONE.
007006  005726                  TST     (SP)+           ;FINALLY THROW AWAY STACK.
007010  000207                  RTS     PC              ;AND RETURN TO THE CALLER.


                        .SBTTL  THE LOADER
```

```
                        ;               THE LOADER
                        ;               --- ------

007012  012737  173000  007010  LOADER: MOV     #DISTOP,JMPADD-2        ;STOP THE GT40 BY INSERTING A "DISTOP" IN THE LI
007020  005003                          CLR     INDEX                   ;RESET THE 8 BIT ASSEMBLER TO THE FIRST CHAR

007022  005005                  L.LD2:  CLR     L.CKSM                  ;CLEAR THE CHECKSUM
007024  004737  167114                  JSR     PC,L.PTR!160000         ;GET A BYTE NOW.
007030  105300                          DECB    L.BYT                   ;IS IT A ONE (HEADER)?
007032  001373                          BNE     L.LD2                   ;NO. WAIT FOR THE ONE.

007034  004737  167114                  JSR     PC,L.PTR!160000         ;YES. SKIP OVER THE NEXT CHARACTER NOW.

007040  004737  167126                  JSR     PC,L.GWRD!160000                ;ASSEMBLE A WORD NOW.
007044  010002                          MOV     L.BYT,L.BC              ;MOVE OVER TO THE COUNTER.
007046  162702  000004                  SUB     #4,L.BC                 ;REDUCE TO ACTUAL DATA COUNT.
007052  022702  000002                  CMP     #2,L.BC                 ;ANY DATA AT ALL?
007056  001433                          BEQ     L.JMP                   ;NO. MUST BE END
007060  004737  167126                  JSR     PC,L.GWRD!160000                ;YES. ASSEMBLE A DATA WORD NOW.
007064  010001                          MOV     L.BYT,L.ADR             ;AND THIS MUST BE THE FIRST ADDRESS.


007066  004737  167114          L.LD3:  JSR     PC,L.PTR!160000         ;GET A BYTE OF DATA NOW.
007072  002006                          BGE     L.LD4                   ;ALL DONE?
007074  105705                          TSTB    L.CKSM                  ;YEP. COUNTER IS MINUS. CHECK CHECKSUM.
007076  001751                          BEQ     L.LD2                   ;CHECKSUM GOOD. GET NEXT COMMAND.


007100  004337  166652          L.BAD:  JSR     SCAN,OUTLIT!160000      ;BAD LOAD INFORM HOST
007104     175     102                  .BYTE   ALTMOD,'B               ;SEND ALTMODE B
007106  000646                          BR      PRESTRT                 ;AND RESTART THE DISPLAY.


007110  110021                  L.LD4:  MOVB    L.BYT,(L.ADR)+          ;INSERT BYTE INTO MEMORY.
007112  000765                          BR      L.LD3                   ;AND GET THE NEXT BYTE.


007114  004737  166664          L.PTR:  JSR     PC,GETB!160000                  ;ASSEMBLE AN 8 BIT CHARACTER NOW.
007120  060005                          ADD     L.BYT,L.CKSM            ;UPDATE THE CHECKSUM NOW.
007122  005302                          DEC     L.BC                    ;DECREMENT THE CHARACTER COUNTER.
007124  000207                          RTS     PC                      ;AND RETURN TO THE CALLER NOW.


007126  004737  167114          L.GWRD: JSR     PC,L.PTR!160000         ;ASSEMBLE A WORD. FIRST GET A CHARACTER
007132  010046                          MOV     L.BYT,-(SP)             ;AND SAVE IT.
007134  004737  167114                  JSR     PC,L.PTR!160000         ;AND THEN GET ANOTHER ONE.
```

```
                                        SWAB    L.BYT                   ;AND THEN REASSEMBLE THE MESS.
                                        BIS     (SP)+,L.BYT             ;WITH THE FEARSOME POWER OF THE II.
                                        RTS     PC                      ;AND RETURN TO THE CALLER.


L.JMP:                                  JSR     PC,L.GWRD!160000                ;ALL DONE WITH THE LOAD. ASSEMBLE
                                        MOV     L.BYT,-(SP)             ;THE STARTING ADDRESS NOW.
                                        JSR     PC,L.PTR!160000         ;AND DON'T FORGET TO CHECKSUM IT.
                                        TSTB    L.CKSM
                                        BNE     L.BAD                   ;A BAD CHECKSUM. ALL IS EVIL.

                                        JSR     SCAN,OUTLIT!160000              ;GOOD CHKSUM,INFORM HOST
                                        .BYTE   ALTMOD,'G               ;WITH ALTMOD G

                                        BIT     #1,(SP)                 ;DO WE WANT TO START EXECUTION?
                                        BEQ     L.JMP1                  ;YES. AWAY WE GO.

L.HALT:                                 HALT                            ;IF NOT, HALT.

L.JMP1:                                 JMP     @(SP)+                  ;IF GO, THEN GO ALREADY. WHEEEE!


                                        .SBTTL  THE SELF TEST
```

```
                                                        ;THIS IS GT40 QUICK TEST
                                                        ;GIVES QUICK VISUAL TEST
                                                        ;OF CONDITION OF MACHINE
                                                        ;WITHOUT READING IN DIAG.
100000          CHAR=100000
104000          SHORTV=104000
110000          LONGV=110000
114000          POINT=114000
120000          GRAPHX=120000
124000          GRAPHY=124000
130000          RELATV=130000

002000          INT0=2000
002200          INT1=2200
002400          INT2=2400
002600          INT3=2600
003000          INT4=3000
003200          INT5=3200
003400          INT6=3400
003600          INT7=3600                               ;BRIGHTEST

000100          LPOFF=100
000140          LPON=140
000020          BLKOFF=20
000030          BLKON=30

000004          LINE0=4
000005          LINE1=5
000006          LINE2=6
000007          LINE3=7

160000          DJMP=160000
164000          DNOP=164000
170000          STATSA=170000
173400          DSTOP=173400                            ;STOP INTERRUPT

000300          LPLITE=300
000200          LPDARK=200
000040          ITALO=40                                ;ITALICS OFF
000060          ITALI=60                                ;"       ON
000004          SYNON=4                                 ;SYNC ON

174000          STATSB=174000

000100          INCR=100                                ;LOAD GRAPH INCR
040000          INTX=40000                              ;INTENSIFY BIT
001777          MAXX=1777                               ;BIGGEST X VECTOR
001377          MAXY=1377                               ;BIGGEST Y VECTOR
020000          MINUSX=20000                            ;THE MINUS BIT
020000          MINUSY=MINUSX
017600          MAXSX=17600                             ;BIGGEST X IN SHORTVEC
000077          MAXSY=77                                ;"      Y IN  "
000100          MINSUY=100                              ;MINUS BIT FOR Y IN SHORTVEC
```

```
2254  007204  012737  167214  172000          MOV     #FILE0!160000,GT40PC            ;START THE GT4C
2255  007212  000001                          WAIT                                    ;AND WAIT
2256
2257  007214  114020                  FILE0:  POINT!BLKOFF                            ;POINT--INVISIBLE
2258  007216  000000                          0
2259  007220  001377                          MAXY
2260
2261  007222  112004                          LONGV!INT0!LINE0                        ;DRAW TOP LINE
2262  007224  041777                          INTX!MAXX
2263  007226  000000                          0
2264
2265  007230  112405                          LONGV!INT2!LINE1
2266  007232  040000                          INTX                                    ;DRAW LINE TO RIGHT
2267  007234  021377                          MINUSX!MAXY
2268
2269  007236  113006                          LONGV!INT4!LINE2
2270  007240  061777                          INTX!MINUSX!MAXX                        ;DRAW BOTTOM LINE
2271  007242  000000                          0
2272
2273  007244  113407                          LONGV!INT6!LINE3
2274  007246  040000                          INTX
2275  007250  001377                          MAXY                                    ;DRAW LINE TO LEFT
2276
2277  007252  114000                          POINT
2278  007254  000400                          400
2279  007256  000500                          500
2280  007260  106200                          SHORTV!INT1
2281  007262  057677                          57677                                   ;+X+Y
2282  007264  106600                          SHORTV!INT3
2283  007266  077677                          77677                                   ;+X-Y
2284  007270  107200                          SHORTV!INT5
2285  007272  077777                          77777                                   ;-X-Y
2286  007274  107600                          SHORTV!INT7
2287  007276  057777                          57777                                   ;-X+Y
2288
2289  007300  114000                          POINT
2290  007302  001400                          1400
2291  007304  000500                          500
2292  007306  133030                          RELATV!INT4!BLKON
2293  007310  057677                          57677                                   ;+X+Y
2294  007312  077677                          77677                                   ;+X-Y
2295  007314  077777                          77777                                   ;-X-Y
2296  007316  057777                          57777                                   ;-X+Y
2297
2298  007320  114000                          POINT
2299  007322  000400                          400
2300  007324  000100                          100
2301  007326  174120                          STATSB!INCR+20                          ;TRY GRAPH MODES
2302  007330  114000                          POINT
2303  007332  001000                          1000
2304  007334  000200                          200
2305
2306  007336  120000                          GRAPHX
2307  007340  001010                          1010
2308  007342  001020                          1020
2309  007344  001030                          1030
```

```
2310  007346  001040                  1040
2311  007350  001050                  1050
2312
2313  007352  114000                  POINT
2314  007354  001000                  1000
2315  007356  001200                  1200
2316
2317  007360  124000                  GRAPHY
2318  007362  001020                  1020
2319  007364  001030                  1030
2320  007366  001040                  1040
2321  007370  001050                  1050
2322  007372  001060                  1060
2323
2324  007374  160000                  DJMP
2325  007376  167214                  FILE0!160000
2326
2327                                  .SBTTL  PAPER TAPE BOOT
```

```
                                        ;
                                        ; PAPER TAPE BOOT
                                        ;
                177550                  HSR=177550                      ;HIGH SPEED READER ADDRESS
                177560                  LSR=177550                      ;LOW SPEED READER ADDRESS
                                        ;       .=ORIGIN+1400
                                        ;
007400  012701  160000          PTBOOT: MOV     #160000,R1              ;SET MEMORY CHECK LIMITS
007404  012702  000004                  MOV     #4,R2                   ;TRAP ADDRESS IS LOC. 4
007410  012703  167500                  MOV     #DEV+4!160000,R3                ;POINTER TO DEVICE ADDRESSES
007414  010712                          MOV     PC,@R2                  ;PRESET TRAP ADDRESS IN LOC. 4
007416  012706  000024                  MOV     #24,SP                  ;STACK SET UP AT SPECIAL ADDRESS
007422  014304                  DEV1:   MOV     -(R3),R4                ;GET DEVICE ADDRESS
007424  005714                          TST     @R4                     ;CHECK AVAILABILITY OF DEVICE
007426  100775                          BMI     DEV1                    ;CHECK DEVICE FOR ERRORS
007430  010712                          MOV     PC,@R2                  ;RESET TRAP ADDRESS AT LOC. 4
007432  012706  000024                  MOV     #24,SP                  ;SPECIAL ADDRESS USED AS MASK LATER
007436  010441                          MOV     R4,-(R1)                ;DO MEM CHK:READER STATUS ADDRESS
                                                                        ;IS MOVED
007440  040601                          BIC     SP,R1                   ;SET R1=X7752,MASK IN SP=24
007442  010111                          MOV     R1,@R1                  ;STORE OWN ADDRESS IN POINTER
007444  011102                  LOOP:   MOV     @R1,R2                  ;GET BYTE POINTER
007446  005214                          INC     @R4                     ;ENABLE READER
007450  105714                          TSTB    @R4                     ;TEST DONE BIT
007452  100376                          BPL     .-2                     ;WAIT UNTIL READY
007454  116412  000002                  MOVB    2(R4),@R2               ;THEN PICK IT UP AND STORE IT
007460  005211                          INC     @R1                     ;BUMP POINTER
007462  120227  000375                  CMPB    R2,#375                 ;STORED JUMP OFFSET?
007466  001366                          BNE     LOOP                    ;NOT YET
007470  105222                          INCB    (R2)+                   ;YES,ALL DONE
007472  000142                          JMP     -(R2)                   ;GO EXECUTE AS BRANCH
                                        ;
                                        ; DEVICE ADDRESSES FOLLOW - DO NOT CHANGE THE ORDER
                                        ;
007474  177560                  DEV:    LSR                             ;LOW SPEED READER
007476  177550                          HSR                             ;HIGH SPEED READER
                                        ;
                                        .SBTTL  CASSETTE BOOT
```

```
                                        ;
                                        ; CASSETTE BOOT
                                        ;
            177500                      TACS=177500                 ;TA-11 CONTROL AND STATUS REGISTER
                                        .=ORIGIN+1500
007500  012700  177500                  TABOOT: MOV     #TACS,R0
007504  005010                                  CLR     (R0)            ;SELECT UNIT #0
007506  010701                          RES:    MOV     PC,R1           ;USE FOR PIC
007510  062701  000052                          ADD     #TABLE-.,R1     ;R1 HOLDS ADDR. OF COMMAND TABLE
007514  012702  000375                          MOV     #375,R2         ;MEMORY PTR. AND DATA FLAG
007520  112103                              1$: MOVB    (R1)+,R3        ;TEST BITS
                                        ;
007522  112110                          LOOP1:  MOVB    (R1)+,(R0)      ;COMMAND FROM TABLE TO TACS
007524  100413                                  BMI     DONE            ;WHEN COMMAND CODE NEG., QUIT
007526  130310                          LOOP2:  BITB    R3,(R0)         ;TEST READY AND T-REG BITS IN TACS
007530  001776                                  BEQ     LOOP2           ;LOOP 'TIL SOMETHING COMES UP
007532  105202                                  INCB    R2              ;ADVANCE MEMORY POINTER
007534  100772                                  BMI     LOOP1           ;IF MINUS, TRY NEXT COMMAND
007536  116012  000002                          MOVB    2(R0),(R2       ;READ DATA INTO MEMORY
007542  120337  000000                          CMPB    R3,@#0          ;FIRST BYTE READ SHOULD BE '240'
007546  001767                                  BEQ     LOOP2           ;IF O.K., GO READ ANOTHER BYTE
007550  000000                          STOP:   HALT                    ;HALT ON ERROR
007552  000755                                  BR      RES             ;RESTART ON CONTINUE
                                        ;
007554  005710                          DONE:   TST     (R0)            ;CHECK FOR ERROR
007556  100774                                  BMI     STOP            ;HALT ON ERROR
007560  005007                                  CLR     PC              ;= 'JMP @#0'
                                        ;
007562  017640                          TABLE:  .WORD   17640           ;.BYTE 240: READY+T-REQ.
                                                                        ;.BYTE 37:  I_BS+READY+GO
007564  002415                                  .WORD   2415            ;.BYTE  15: SFB+GO
                                                                        ;.BYTE   5: READ+GO
007566  112024                                  .WORD   112024          ;.BYTE  24: READ+ILBS
                                                                        ;.BYTE 224: READ+ILBS+E.O.TABLE
007570  000000  000000                          .WORD   0,0             ;THESE ARE FILLER WORDS
007574  167500                                  .WORD   TABOOT!160000           ;POWER UP VECTOR AND PRIORITY
007576  000340                                  .WORD   340             ;


                                        .SBTTL  MR11-DB BOOT
```

```
2406                                    ;MR11-DB BULK STORAGE PROGRAM LOADER LISTING
2407
2408                                    :       .=ORIGIN+1600                   ;KEEP TRACK OF ORIGIN
2409
2410  007600  010702                    RF11:   MOV PC,R2               ;FIXED HEAD DISK (256 KW)
2411  007602  000451                            BR OTHER
2412  007604  177462                            177462
2413  007606  000005                            5
2414
2415  007610  010702                    RK11:   MOV PC,R2               ;MOVING HEAD DISK (CARTRIDGE)
2416  007612  000445                            BR OTHER
2417  007614  177406                            177406
2418  007616  000005                            5
2419
2420
2421  007620  010702                    TC11:   MOV PC,R2
2422  007622  000417                            BR TAPES
2423  007624  177344                            177344                  ;ADDRESS OF WORD COUNT
2424  007626  000005                            5                       ;LAST COMMAND
2425  007630  004003                            4003                    ;FIRST COMMAND
2426  007632  100000                            100000                  ;DONE MASK
2427  007634  024000                            24000                   ;ERROR MASK
2428
2429
2430  007636  010702                    TM11:   MOV PC,R2
2431  007640  000410                            BR TAPES
2432  007642  172524                            172524                  ;ADDRESS OF BYTE COUNT
2433  007644  060003                            60003                   ;LAST COMMAND
2434  007646  060011                            60011                   ;FIRST COMMAND
2435  007650  000200                            200                     ;DONE MASK
2436  007652  100000                            100000                  ;ERROR MASK
2437
2438
2439  007654  010702                    RP11:   MOV PC,R2               ;MOVING HEAD DISK (PACK)
2440  007656  000423                            BR OTHER
2441  007660  176716                            176716
2442
2443
2444  007662  000005                    TAPES:  RESET
2445  007664  010200                            MOV R2,R0               ;GET THE ADDRESS OF THE BRANCH
2446  007666  005720                            TST (0)+                ;R0 TO POINT AT LAST COMMAND
2447  007670  012001                            MOV (0)+,R1             ;GET THE WORD COUNT ADDRESS
2448  007672  005311                            DEC (1)                 ;SET UP FOR ADVANCE 1 RECORD
2449  007674  005720                            TST (0)+                ;MOVE R0 TO FIRST COMMAND
2450  007676  012041                            MOV (0)+,-(1)           ;COMMAND WORD TO COMMAND REG.
2451  007700  031011                            BIT (0),(1)             ;LOOK FOR DONE INDICATORS
2452  007702  001776                            BEQ .-2                 ;NONE SET, TRY AGAIN
2453  007704  005720                            TST (0)+                ;DONE FIRST COMMAND, CHECK FOR ERROR
2454  007706  031041                            BIT (0),-(1)            ;LOOK FOR SET ERROR BITS
2455  007710  001406                            BEQ OTHER               ;NO ERRORS - TRY THE READ
2456  007712  000112                    AGAIN:  JMP (2)                 ;RERUN FOR ERRORS
2457
2458
2459  007714  167600                    RFVEC:  RF11!160000                     ;RF11 POWER UP VECTOR
2460  007716  000340                            340
2461
```

```
  2462  007720  010702                RC11:   MOV PC,R2               ;FIXED HEAD DISK (64KW)
  2463  007722  000401                        BR OTHER
  2464  007724  177450                        177450                  ;ADRS OF WORD COUNT (COMMAND-2
  2465                                                                  ;COMMAND WORD (5) IS THE RESET
  2466
  2467  007726  000005                OTHER:  RESET
  2468  007730  010200                        MOV R2,R0               ;R0 TO POINT AT WORD COUNT ADRS
  2469  007732  005720                        TST (0)+                ;POINT TO ADDRESS
  2470  007734  012001                        MOV (0)+,R1             ;WORD COUNT ADDRESS TO R1
  2471  007736  012711  177000                MOV #-1000,(1)          ;LOAD WORD COUNT
  2472  007742  011041                        MOV (0),-(1)            ;COMMAND TO COMMAND REGISTER
  2473  007744  032711  100200                BIT #100200,(1)         ;CHECK FOR ERROR OR DONE
  2474  007750  001775                        BEQ .-4                 ;IF NEITHER, KEEP LOOKING
  2475  007752  100757                        BMI AGAIN               ;ERROR, TRY AGAIN
  2476  007754  005007                        CLR PC
  2477
  2478  007756  000000                        0                       ;FILLER
  2479  007760  167610                RKVEC:  RK11!160000                     ;RK POWER UP VECTOR
  2480  007762  000340                        340
  2481  007764  167720                RCVEC:  RC11!160000                     ;RC POWER UP VECTOR
  2482  007766  000340                        340
  2483  007770  167654                RPVEC:  RP11!160000                     ;RP POWER UP VECTOR
  2484  007772  000340                        340
  2485  007774  167620                TCVEC:  TC11!160000                     ;TC11 POWER UP VECTOR
  2486  007776  000340                        340
  2487
  2488                                        .SBTTL  ROM VERSION 1 VALUES
```

```
                                        .DSABL  AMA

                                ;DATA PATTERN STORED IN THE GT40 BOOTSTRAP VERSION 1

                                ;       ************ THIS IS A IMAGE LISTING OF THE GT40 (VT40) BOOTSTRAP ********
                                ;
                                ;       THE DATA IS A MIRROR IMAGE OF THAT IN THE BOOTSTRAP ROMS
                                ;       ONLY THE ADDRESS FIELD IS CHANGED

                                ;BOOTVT.SO9  5/2/72      <SPECIAL>


                                ;       VT-40 BOOTSTRAP LOADER, VERSION SO9, RELEASE R01, 5/2/72

                                ;       COPYRIGHT 1972, DIGITAL EQUIPMENT CORPORATION.
                                ;       146 MAIN STREET
                                ;       MAYNARD, MASSACHUSSETTS
                                ;                               01754


                                ;       WRITTEN BY JACK BURNESS, SENIOR SYSTEMS ARCHITECT!



                                ;       THIS ROUTINE IS INTENDED TO BE LOADED IN THE ROM PORTION OF THE VT-40.


                                ;       REGISTER DEFINITIONS:

000000                          R0=%0
000001                          R1=%1
000002                          R2=%2
000003                          R3=%3
000004                          R4=%4
000005                          R5=%5
000006                          R6=%6
000007                          R7=%7

000006                          SP=R6
000007                          PC=R7

000000                          RET1=R0                 ;RETURN OF VALUE REGISTER.
000001                          INP1=R1                 ;ARGUMENT FOR CALLED FUNCTION
000002                          INP2=R2                 ;SECOND ARGUMENT.
000003                          WORK1=R3                ;FIRST WORK REGISTER.
000004                          WORK2=R4                ;SECOND WORKING REGISTER.
000005                          SCR1=R5                 ;SCRATCH REGISTER.

000003                          LCKSM=WORK1             ;OVERLAPPING DEFINITIONS FOR LOADER PORTION.
000000                          LBYT=RET1
000005                          LBC=SCR1
000001                          LADR=INP1
```

```
                036000                  COREND=36000                    ;FIRST LOCATION OF NON-CORE.
                166000                  ROMORG=166000                   ;WHERE THE ROM PROGRAM SHOULD GO.

                000000                  STARTX=0                        ;WHERE TO START DISPLAYING THE X POSITIONS.
                001360                  STARTY=1360                     ;WHERE TO START DISPLAYING THE Y.


                012000                  VT40PC=172000-150000                    ;VT40 PROGRAM COUNTER.
                027560                  KBDIS=27560                     ;TTY INPUT STATUS.
                025614                  P10OS=25614                     ;PDP-10 OUTPUT STATUS.
                025610                  P10IS=25610                     ;PDP-10 INPUT STATUS.

                027562                  KBDIB=KBDIS+2                   ;TTY INPUT BUFFER.
                025612                  P10IB=P10IS+2                   ;PDP-10 INPUT CHARACTER.
                025616                  P10OB=P10OS+2                   ;PDP-10 OUTPUT BUFFER.


                045776                  P10OC=COREND-2+10000                    ;CHARACTER TO BE SENT TO THE PDP-10
                045772                  P10IC=P10OC-4                   ;INPUT CHARACTER FROM IO PLUS ONE SAVE CHARACTER
                015770                  STKSRT=P10IC-2-30000                    ;FIRST LOCATION OF STACK.


                160000                  JMPDIS=160000                   ;THE VT-40 DISPLAY JUMP INSTRUCTION.


                000024                  PWRFAL=24                       ;POWER FAIL RESTART LOCATION.



                016000                          .=16000
                                        ;       .=ROMORG                ;SET THE ORIGIN NOW!!!!



016000  012705  000026          STARTA: MOV     #PWRFAL+2,SCR1          ;PICK UP POINTER TO P.F. STATUS.
016004  005015                          CLR     @SCR1                   ;CLEAR IT OUT TO BE SURE.
016006  010745                          MOV     PC,-(SCR1)              ;SET UP THE RESTART LOCATION.

016010  000005                          RESET                           ;RESET THE BUS.

016012  012767  000007  007570          MOV     #7,P10IS                ;INITIALIZE PDP-10 INPUT
016020  012767  000001  011532          MOV     #1,KBDIS                ;INITIALIZE TTY INPUT.
016026  012767  000201  007560          MOV     #201,P10OS              ;INITIALIZE PDP-10 OUTPUT.
```

```
016034  012706  015770      RSTPT:  MOV     #STKSRT,SP                      ;SET UP THE STACK NOW!
016040  005001                      CLR     LADR                            ;CLEAR ADDRESS POINTER.
016042  012702  160000              MOV     #JMPDIS,INP2                    ;PLACE A DISPLAY JUMP INSTRUCTION IN A REGISTER.
016046  010221                      MOV     INP2,(LADR)+                    ;MOVE IT TO LOCATION 0.
016050  012711  166756              MOV     #DISPRG+150000,(LADR)                   ;MOVE ADDRESS POINTER INTO 2.
016054  012701  000030              MOV     #PWRFAL+4,LADR                  ;SET UP WHERE WE WILL STORE CHARACTERS.
016060  005000                      CLR     RET1                            ;PREPARE TO INSERT A ZERO CHARACTER.
016062  004767  000022              JSR     PC,DOCHAR                       ;INSERT IT NOW.
016066  005067  003706              CLR     VT40PC                          ;CLEAR THE DISPLAY PROGRAM COUNTER AND START.

016072  004767  000210      MAJOR:  JSR     PC,GTCHR                        ;GT A CHARACTER NOW.
016076  000240                      NOP
016100  000240                      NOP
016102  000240                      NOP
016104  012746  156072              MOV     #MAJOR+150000,-(SP)                     ;INSERT IN DISPLAY BUFFER NOW.

016110  010105              DOCHAR: MOV     LADR,SCR1                       ;GT CURRENT BUUFER POSITION NOW.
016112  022525                      CMP     (SCR1)+,(SCR1)+                 ;BYPASS CURRENT DISPLAY JUMP.
016114  005025                      CLR     (SCR1)+                         ;CLEAR FUTURE ADDRESS FOR JUMP.
016116  010225                      MOV     INP2,(SCR1)+                    ;STICK IN TEMPORARY JUMP WHILE WE REPLACE CURREN
016120  005015                      CLR     (SCR1)                          ;A DISPLAY JUMP TO ZERO.
016122  005011                      CLR     (LADR)                          ;NOW REPLACE CURRENT DISPLAY JUMP BY THE CHARACT
016124  050021                      BIS     RET1,(LADR)+                    ;IT'S DONE THIS WAY TO WASTE 2 CYCLES.
016126  010211                      MOV     INP2,(LADR)                     ;TO AVOID TIMING PROBLEMS WITH THE VT40.
016130  000207                      RTS     PC                              ;AND FINALLY RETURN.

016132  004767  000124      GT8:    JSR     PC,GTSIX                        ;GT SIX BITS NOW.
016136  010046                      MOV     RET1,-(SP)                      ;SAVE THE CHARACTER NOW.
016140  000401                      BR      GTP84                           ;BYPASS THE 8'ER
016142  005002              GT84:   CLR     INP2                            ;RESET THE MAGIC REGISTER NOW.
016144  005722              GTP84:  TST     (INP2)+                         ;INCREMENT WHERE TO GO.
016146  066207  166250              ADD     GT8TB+150000(INP2),PC                   ;UPDATE PC NOW.

        016152              GT8P=.

016152  004767  000104      GT81:   JSR     PC,GTSIX                        ;GT A CHARACTER NOW.
016156  010004                      MOV     RET1,WORK2                      ;SAVE FOR A SECOND.
016160  006300                      ASL     RET1
```

```
2657  016162  006300                  ASL     RET1                    ;SHIFT TO LEFT OF BYTE
2658  016164  106300                  ASLB    RET1
2659  016166  106116                  ROLB    @SP                     ;PACK THEM IN.
2660  016170  106300                  ASLB    RET1
2661  016172  106116                  ROLB    @SP                     ;A GOOD 8 BIT THING.
2662  016174  012600                  MOV     (SP)+,RET1              ;POP AND RETURN NOW.
2663  016176  000207                  RTS     PC
2664
2665  016200  006300          GT82:   ASL     RET1                    ;WORST CASE. SHIFT 4
2666  016202  006300                  ASL     RET1
2667  016204  106300                  ASLB    RET1
2668  016206  106104                  ROLB    WORK2
2669  016210  106300                  ASLB    RET1
2670  016212  106104                  ROLB    WORK2
2671  016214  106300                  ASLB    RET1
2672  016216  106104                  ROLB    WORK2
2673  016220  106300                  ASLB    RET1
2674  016222  106104                  ROLB    WORK2
2675  016224  010400                  MOV     WORK2,RET1
2676  016226  012604                  MOV     (SP)+,WORK2
2677  016230  000207                  RTS     PC
2678
2679  016232  006100          GT83:   ROL     RET1
2680  016234  006100                  ROL     RET1
2681  016236  006004                  ROR     WORK2
2682  016240  106000                  RORB    RET1
2683  016242  006004                  ROR     WORK2
2684  016244  106000                  RORB    RET1                    ;FINAL CHARACTER ASSEMBLED.
2685  016246  005726                  TST     (SP)+                   ;FUDGE STACK.
2686  016250  000207                  RTS     PC                      ;AND RETURN NOW.
2687
2688          016250          GTBTB   =       .-2                     ;PUSH ZERO CONDITION BACK INTO NEVER-NEVER LAND.
2689
2690  016252  000000                  .WORD   GT81-GT8P
2691  016254  000026                  .WORD   GT82-GT8P
2692  016256  000060                  .WORD   GT83-GT8P
2693  016260  177770                  .WORD   GT84-GT8P
2694
2695
2696  016262  004767  000020  GTSIX:  JSR     PC,GTCHR
2697  016266  020027  000040          CMP     RET1,#40
2698  016272  002546                  BLT     LBAD
2699  016274  020027  000137          CMP     RET1,#137
2700  016300  003143                  BGT     LBAD
2701  016302  000207                  RTS     PC
2702
2703
2704
2705  016304  005726          GTCHP:  TST     (SP)+                   ;UPDATE THE STACK.
2706
2707  016306  012700  015772  GTCHR:  MOV     #P1OIC-30000,RET1               ;SET UP POINTER TO THE INPUT CHARACTER.
2708  016312  004767  000064  GTCHL:  JSR     PC,CHECK
2709  016316  005710                  TST     @RET1                   ;ANY CHARACTERS THERE?
2710  016320  001774                  BEQ     GTCHL
2711  016322  011046                  MOV     @RET1,-(SP)             ;PUSH THE CHAR ON THE STACK.
2712  016324  005020                  CLR     (RET1)+                 ;CLEAR THE CHAR GOT FLAG NOW.
```

```
 2713  016326  042716  177600              BIC     #-200,(SP)              ;CLEAR AWAY PARITY NOW.
 2714  016332  001764                      BEQ     GTCHP                   ;IF ZERO, GT ANOTHER
 2715  016334  022716  000177              CMP     #177,(SP)
 2716  016340  001761                      BEQ     GTCHP                   ;ALSO IGNORE RUBOUTS.
 2717  016342  022710  000175              CMP     #175,@RET1              ;WAS IT A "175"
 2718  016346  001007                      BNE     GTNP                    ;NOPE.
 2719  016350  011610                      MOV     (SP),@RET1              ;YEP. RESET IN CASE OF ABORT.
 2720  016352  021027  000122              CMP     @RET1,#122              ;IS IT AN R
 2721  016356  001626                      BEQ     RSTRT                   ;YEP. RESTART
 2722  016360  021027  000114              CMP     @RET1,#114              ;IS IT AN L
 2723  016364  001455                      BEQ     LOAD                    ;YEP. LOAD.
 2724
 2725  016366  011610              GTNP:   MOV     (SP),@RET1              ;NOW DO THE FDUGING.
 2726  016370  012600                      MOV     (SP)+,RET1
 2727  016372  020027  000175              CMP     RET1,#175
 2728  016376  001743                      BEQ     GTCHR                   ;IF ALTMODE, LOOP
 2729  016400  000207                      RTS     PC
 2730
 2731
 2732
 2733
 2734
 2735
 2736
 2737
 2738  016402  005767  027370      CHECK:  TST     P100C                   ;DO WE WANT TO OUTPUT?
 2739  016406  001410                      BEQ     CHECK1                  ;NO.
 2740  016410  105767  007200              TSTB    P100S                   ;WE DO. IS THE IO READY?
 2741  016414  100005                      BPL     CHECK1                  ;NOT QUITE.
 2742  016416  016767  027354  007172      MOV     P100C,P100B             ;IT'S READY. SEND THE CHARACTER.
 2743  016424  005067  027346              CLR     P100C                   ;AND THE SAVED CHARACTER.
 2744
 2745  016430  105767  011124      CHECK1: TSTB    KBDIS                   ;HEY, IS THE KEYBOARD READY?
 2746  016434  100014                      BPL     CHECK3                  ;NOPE. NO LUCK.
 2747  016436  116746  011120              MOVB    KBDIB,-(SP)             ;YEP. SAVE THE CHARACTER NOW.
 2748  016442  012767  000001  011110      MOV     #1,KBDIS                ;AND REENABLE THE COMMUNICATIONS DEVICE.
 2749
 2750  016450  004767  177726      CHECK2: JSR     PC,CHECK                ;IS THE OUTPUT READY?
 2751  016454  005767  027316              TST     P100C
 2752  016460  001373                      BNE     CHECK2                  ;IF NOT, WAIT TILL DONE.
 2753  016462  012667  007130              MOV     (SP)+,P100B             ;AND THEN SEND OUT THE CHARACTER.
 2754
 2755
 2756  016466  105767  007116      CHECK3: TSTB    P101S                   ;IS THE IO TALKING TO ME.
 2757  016472  100011                      BPL     CHECK4                  ;NOPE. EXIT.
 2758  016474  116767  007112  027270      MOVB    P101B,P101C             ;GT THE CHARACTER NOW.
 2759  016502  052767  177400  027262      BIS     #-400,P101C             ;MAKE SURE IT'S NONE ZERO.
 2760  016510  012767  000007  007072      MOV     #7,P101S                ;REINITIALIZE COMMUNICATION LINE.
 2761
 2762  016516  000207              CHECK4: RTS     PC                      ;AND RETURN.
 2763
 2764
 2765
 2766
 2767
 2768
```

```
                                        ;       THE "  L  O  A  D  E  R

016520  005002                  LOAD:   CLR"    INP2                    ;RESET TO FIRST 8 BIT CHARACTER.
016522  012712  172000                  MOV     #172000,(INP2)          ;AND ALSO CLEVERLY STOP THE VT40.
016526  012706  015770                  MOV "   #STKSRT,SP              ;RESET STACK POINTER NOW.

016532  005003                  LLD2:   CLR     LCKSM                   ;CLEAR THE CHECKSUM
016534  004767  000070                  JSR     PC,LPTR         ;GT A BYTE NOW.
016540  105300                          DECB    LBYT                    ;IS IT ONE?
016542  001373                          BNE     LLD2                    ;NOPE. WAIT AWHILE
016544  004767  000050                  JSR     PC,LPTR         ;YEP. GT NEXT CHARACTER.

016550  004767  000072                  JSR     PC,LGWRD                ;GT A WORD.
016554  010005                          MOV     LBYT,LBC                ;GT THE COUNTER NOW.
016556  162705  000004                  SUB     #4,LBC                  ;CHOP OFF EXTRA STUFF.
016562  022705  000002                  CMP     #2,LBC                  ;NULL?
016566  001437                          BEQ     LJMP                    ;YEP. MUST BE END.
016570  004767  000052                  JSR     PC,LGWRD                ;NOPE. GT THE ADDRESS.
016574  010001                          MOV     LBYT,LADR               ;AND REMEMBER FOR OLD TIMES SAKE.

016576  004767  000026          LLD3:   JSR     PC,LPTR         ;GT A BYTE (DATA)
016602  002010                          BGE     LLD4                    ;ALL DONE WITH THE COUNTER?
016604  105703                          TSTB    LCKSM                   ;YEP. GOOD CHECK SUM?
016606  001751                          BEQ     LLD2                    ;NOPE. LOAD ERROR.

016610  012700                  LBAD:   MOV     (PC)+,RET1              ;SEND OUT SOME CHARACTERS NOW.
                                :       .BYTE   175,102                 ;"CTRL BAD"
016612      102     000175              .BYTE   102,175                 ;"BAD CTRL"
016614  004767  000110                  JSR     PC,SENDIT
016620  000167  177210                  JMP     RSTRT

016624  110021                  LLD4:   MOVB    LBYT,(LADR)+            ;PLACE THE BYTE IN CORE.
016626  000763                          BR      LLD3                    ;GT ANOTHER ONE.

016630  004767  177276          LPTR:   JSR     PC,GT8                  ;GT 8 BITS NOW.
016634  060003                          ADD     LBYT,LCKSM              ;UPDATE CHECKSUM
016636  042700  177400                  BIC     #177400,LBYT            ;CLEAN UP THE BYTE NOW.
016642  005305                          DEC     LBC                     ;UPDATE THE COUNTER.
016644  000207                          RTS     PC                      ;RETURN NOW.

016646  004767  177756          LGWRD:  JSR     PC,LPTR         ;GT A CHARACTER.
016652  010046                          MOV     LBYT,-(SP)              ;SAVE FOR A SECOND.
016654  004767  177750                  JSR     PC,LPTR         ;GT ANOTHER CHARACTER.
016660  000300                          SWAB    LBYT                    ;NOW ASSEMBLE THE WORD.
016662  052600                          BIS     (SP)+,LBYT              ;AND RETURN WITH A 16 BITER.
016664  000207                          RTS     PC
```

```
2825
2826  016666  004767  177754          LJMP:   JSR     PC,LGWRD                ;GT A WORD
2827  016672  010046                          MOV     LBYT,-(SP               ;SAVE ON THE STACK.
2828  016674  004767  177730                  JSR     PC,LPTR         ;GT A CHARCTER.
2829  016700  105703                          TSTB    LCKSM                   ;IS IT ZERO?
2830  016702  001342                          BNE     LBAD                    ;YEP. WHAT CRAP.
2831  016704  032716  000001                  BIT     #1,(SP)                 ;IS IT ODD?
2832  016710  001406                          BEQ     LJMP1                   ;YEP. START PROGRAM GOING NOW.
2833  016712  012700                          MOV     (PC)+,RET1              ;TELL PDP-10 WE'VE LOADED OK.
2834                                  :       .BYTE   175,107                 ;"CTRL GOOD"
2835  016714     107     175                  .BYTE   107,175                 ;"GOOD CTRL"
2836  016716  004767  000006                  JSR     PC,SENDIT
2837  016722  000000                          HALT
2838  016724  000776                          BR      .-2
2839
2840  016726  000136          LJMP1:  JMP     @(SP)+                  ;AND AWAY WE GO.
2841
2842
2843
2844
2845
2846
2847
2848
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858  016730  004767  177446          SENDIT: JSR     PC,CHECK                ;POLL THE OUTPUT DEVICE NOW.
2859  016734  005767  027036                  TST     P100C                   ;OUTPUT CLEAR?
2860  016740  001373                          BNE     SENDIT                  ;NOPE. LOOP AWHILE LONGER.
2861  016742  010067  006650                  MOV     RET1,P100B              ;SEND OUT THE CHARACTER.
2862  016746  105000                          CLRB    RET1                    ;CLEAR THE BYTE.
2863  016750  000300                          SWAB    RET1                    ;AND SWAP THEM NOW.
2864  016752  001366                          BNE     SENDIT                  ;IF NOT EQUAL, REPEAT.
2865  016754  000207                          RTS     PC
2866
2867
2868
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2880
```

```
2881
2882
2883
2884                            ;               THIS IS THE INITIALIZING VT40 PROGRAM WHICH WILL
2885                            ;               JUMP TO THE PROGRAM AFTER THE POWER FAIL LOCATIONS
2886                            ;               WHICH WILL JUMP TO ZERO WHICH WILL JUMP BACK TO HERE.
2887
2888
2889
2890
2891  016756  170256           DISPRG: .WORD   170256          ;LOAD STATUS REGISTER FOR NORMAL OPERATION.
2892  016760  115124                   .WORD   115124          ;SET POINT MODE, "NORMAL".
2893  016762  000000                   .WORD   STARTX          ;X COORDINATE
2894  016764  001360                   .WORD   STARTY          ;Y COORDINATE
2895  016766  100000                   .WORD   100000          ;SET CHARACTER MODE.
2896  016770  160000                   .WORD   JMPDIS          ;THEN JUMP TO THE POWERFAIL LOCATION.
2897  016772  000030                   .WORD   PWRFAL+4        ;TO DISPLAY USERS CHARACTERS.
2898  016774  000000                   .WORD   0
2899  016776  000000                   .WORD   0
2900
2901
2902
2903                            ;STARTING FROM HERE TO THE TOP OF MEMORY
2904                            ; A BACKGROUND WORSE CASE NOISE TASK WILL BE EXECUTED
2905
2906  017000  000000           BUFFER: 0.
2907
2908          000001                   .END
```

```
AGAIN   007712          2456#   2475
ALTMOD= 000175          1664#   1982    2141    2174
BACK    002636           796#    817
BACK1   002654           900#    805     810
BACK2   002670           803     807#
BELL    006250          1803    1835#
BLIMIT= 007000          1657#   1660    1749    1770    1854    1873    1920    1924
BLKOFF= 000020           187#   1212    1236    1265    2221#   2257
BLKON = 000030           187#   1200    1224    1245    2222#   2292
BSTART= 001000          1656#   1856    1876    1893    1923
BUFFER  017000           629     815    2906#
BUFF1   005422           533     704    1454#
BUFF2   005516           582     584     725#    731#   1467#
BUFF3   005612           586     588     754#    760#   1487#
CHAR  = 100000           127#   1291    1309    1326    1344    1362    1380    1451    1464    1484    1505    1520    1608#
CHECK   016402          2708    2738#   2750    2858
CHECK1  016430          2739    2741    2745#
CHECK2  016450          2750#   2752
CHECK3  016466          2746    2756#
CHECK4  016516          2757    2762#
CORENO= 036000          2546#   2563
CORSTR= 000004          1659#   1718
CORTST  002004           628#
CR      006206          1812#
CRLF  = 005015          1663#   1752
DEV     007474          2337    2362#
DEV1    007422          2340#   2342
DISJMP= 160000          1666#   1884    1922    1923    1924    1940
DISPRG  016756          2608    2891#
DISTOP= 173000          1667#   1991    2113
DJMP  = 160000           187#   1515    1526    2229#   2324
DLADC   001006           463#    544
DLBRL   001012           465#    594     596
DLICSR  001156           497#    531#    716
DLIDBR  001160           495#    723
DLIN    002326           593     716#
DLINB   002412           728     730#
DLIVT   001202           511#    593#
DLIVT1  001204           512#    594#
DLOCSR  001162           499#    530#    696
DLODBR  001164           484#    485#    500#    533#    710#
DLOUT   002244           595     696#
DLOUTA  002262           703#
DLOUTB  002314           707     710#
DLOVT   001206           513#    595#
DLOVT1  001210           514#    596#
DLVCT   001010           464#    552
DL11IB= 175612          1645#   1646    1961
DL11IS= 175610          1644#   1645    1711#   1959    1962#
DL11OB= 175616          1647#   1969#   2016#   2017#
DL11OS= 175614          1646#   1647    1714#   1732    1734#
DNOP  = 164000           187#   1289    1304    1508    1509    1510    1511    1512    1513    1514    2230#
DOCHAR  016110          2611    2620#
DOCORE  001720           486     611#
DONE    007554          2379    2390#
DSTOP = 173400           187#   1525    2232#
```

```
DSWR  = 177570            449*   564
ENDCOR  006036           1721*  1722
ERRVEC= 000004            448*   562    563*   570*
FF      006256           1808   1838*
FFLOOP  006262           1833   1840*  1843
FILEC   007214           2254   2257*  2325
FILEDA  005722            580*   683*   687*  1516*
FILEDB  005724            687   1517*
FILEDC  005752            580    683   1516   1525*
FILEDD  003012            529    843*  1527
GETCHR  006564           1791   1981*  2003
GETDL   006516           1959*  1964   1968   1970   1981   1985
GETDL1  006546           1960   1967*
GETEXT  006650           1983   1989   2009*
GETSIX  006630           2003*  2041   2056
GETB    006664           2041*  2150
GETBTB  006700           2044   2046*
GETB1   006712           2046   2056*
GETB2   006740           2047   2068*
GETB3   006772           2048   2083*
GETB4   006706           2052*
GRAPHX= 120000            187*  1040   2206*  2306
GRAPHY= 124000            187*   876   2207*  2317
GRPINC  003074            581*   679*   680    682*   874*
GTADD   001000            459*   542
GTBPL   001004            461*   574    576    578
GTBUSE  006424           1893*  1916
GTCHL   016312           2708*  2710
GTCHP   016304           2705*  2714   2716
GTCHR   016306           2614   2696   2707*  2728
GTDLYC  001314            540*   579*   676*   578*
GTDNE1  001170            503*   574*
GTDONE  001166            502*   573*
GTLPEN  002220            575    687*
GTLPH   001172            505*   575*
GTLPH1  001174            506*   576*
GTMP    016366           2718   2725*
GTPC    001146            492*   529*   541    684*   688*
GTPB4   016144           2647   2649*
GTSHIF  002236            577    691*
GTSIX   016262           2645   2654   2696*
GTSOTM  001176            508*   577*
GTSOT1  001200            509*   578*
GTSR    001150            493*   671    818*
GTSTOP  002130            573    671*
GTST1   002202            677    681    683*
GTVCT   001002            460*   546
GTXPOS  001152            494*
GTYPOS  001154            495*
GT40PC= 172000           1652*  1653   1893*  1927   2254*
GT40SR= 172002           1653*  1835*
GTB     016132           2645*  2813
GTBP  = 016152           2652*  2690   2691   2692   2693
GTBTB = 016250           2650   2688*
GTB1    016152           2654*  2690
GTB2    016200           2665*  2691
```

```
GT83    016232          2679*   2692
GT84    016142          2648*   2693
HEADER  006474          1883    1922    1928    1934*
HSR   = 177550          2331*   2363
IMAGE   001020           469*    474*    480*    483*    797
INCR  = 000100           187*    581     680     682     874    2243*   2301
INGT    001474           554     573*
INITGT  001316           473     541*
INSERT  006350          1816    1824    1962    1870*
INSRTL  006406          1874    1877    1882*
INSRTX  006422          1872    1887*
INTD    001566           583*    585
INTE    001602           587*    589
INTX  = 040000           187*    851     854     857     860     869    1035    1201    1202    1203    1204    1205    1206
                        1207    1208    1213    1214    1215    1216    1217    1218    1219    1220    1225    1226    1227
                        1228    1229    1230    1231    1232    1237    1238    1239    1240    1241    1242    1243    1244
                        1249    1251    1253    1255    1257    1259    1261    1263    1269    1271    1273    1275    1277
                        1279    1281    1283    1401    1407    1413    1419    1425    1431    1437    1443    2244*   2262
                        2266    2270    2274
INTO  = 002000           187*    1442    2210*   2261
INT1  = 002200           187*    1436    2211*   2280
INT2  = 002400           187*    1430    2212*   2265
INT3  = 002600           187*    1424    2213*   2282
INT4  = 003000           187*     850    1418    1448    2214*   2269    2292
INT5  = 003200           187*    1412    2215*   2284
INT6  = 003400           187*    1406    2216*   2273
INT7  = 003600           187*    1400    2217*   2286
ITAL0 = 000040           187*    1290    1322    1358    1394    2236*
ITAL1 = 000060           187*    1305    1340    1376    2237*
JMPADD= 007012          1660*    1850    1861*   1940    1991*   2113*
JMPDIS= 160000          2568*    2606    2896
KBCHR   002626           752*    753*    754     764     784*
KBDIB = 027562          1650*    1969    2558*   2747
KBDIS = 027560          1649*    1650    1967    2554*   2558    2599*   2745    2748*
KBIN    002440           597     744*
KPNT    002624           536*    751     755*    756     758*    759     783*
KRBBRL  001226           524*    598
KRBVT   001222           521*    597*
KRBVT1  001224           522*    598*
LBAD    016610          2698    2700    2804*   2830
LF      006300          1806    1347*
LFLOOP  006310          1852*   1855    1857
LFOUND  006330          1853    1859*
LFSUB   006304          1841    1850*
LGWRD   016646          2791    2796    2819*   2826
LINE0 = 000004           187*    850     865    1400    2224*   2261
LINE1 = 000005           187*    853    2225*   2265
LINE2 = 000006           187*    856    2226*   2269
LINE3 = 000007           187*    859    2227*   2273
LJMP    016666          2795    2826*
LJMP1   016726          2832    2840*
LLD2    016532          2785*   2788    2802
LLD3    016576          2799*   2811
LLD4    016624          2800    2810*
LOAD    016520          2723    2781*
LOADER  007012          1987    2113*
```

```
LOADRO  001400            543     545     547     553     555*
LOGICA  002772            831*
LONGV = 110000            187*    850     853     856     859     866    1034    1248    1263    1400    1406    1412    1418
                         1424    1430    1436    1442    2204*   2261    2265    2269    2273
LOOP    007444           2349*   2356
LOOP1   007522           2378*   2383
LOOP2   007526           2380*   2381    2386
LPDARK= 000200            187*   2235*
LPLITE= 000300            187*    846    2234*
LPOFF = 000100            187*   1265    2219*
LPON  = 000140            187*    843    1286    2220*
LPTR    016630           2786    2789    2799    2813*   2819    2821    2828
LSR   = 177560           2332*   2362
L.BAD   007100           2005    2007    2140*   2171
L.GWRD  007126           2125    2130    2157*   2167
L.HALT  007200           2179*
L.JMP   007146           2129    2167*
L.JMP1  007202           2177    2181*
L.LD2   007022           2118*   2121    2137
L.LD3   007066           2134*   2146
L.LD4   007110           2125    2145*
L.PTR   007114           2119    2123    2134    2150*   2157    2159    2169
MAJOR   016072           2614*   2618
MAXSX = 017600            187*   2249*
MAXSY = 000077            187*   2250*
MAXX  = 001777            187*    851     857    2245*   2262    2270
MAXY  = 001377            187*    845     855     861    1288    1308    1325    1343    1361    1379    2246*   2259    2267
                         2275
MEMDAT  002062            651*    652*    657     660
MEMEMD  002124            634     663*
MEMTST  002054            632     649*
MINSUY= 000100            187*   1206    1207    1208    1218    1219    1220    1230    1231    1232    1242    1243    1244
                         2251*
MINUSX= 020000            187*    855     857    1204    1205    1206    1216    1217    1218    1228    1229    1230    1240
                         1241    1242    1255    1257    1259    1260    1262    1264    1275    1277    1279    1280    1282
                         1284    2247*   2248    2267    2270
MINUSY= 020000            187*   2248*
NORMAL  006212           1795    1804    1816*
NOTHER  006042           1719    1725*
NUMLIN= 000040           1661*   1749    1750    1838    1924
NXTCHR  006132           1791*   1793    1799    1818    1828    1836    1844    1847
OCTA    005677            590*    591*    592*    766*    768*    770*   1507*
ORIGIN= 166000           1642*
OTHER   007726           2411    2416    2440    2455    2463    2467*
OUTLIT  006652           1715    2016*   2140    2173
OVER    002630            489     787*    826
PCOUNT  003010            787*    816*    839*
POINT = 114000            187*    843     865     871    1031    1037    1197    1209    1221    1233    1245    1265    1296
                         1306    1323    1341    1359    1377    1397    1403    1409    1415    1421    1427    1433    1439
                         1448    1461    1481    1502    1517    2205*   2257    2277    2289    2298    2302    2313
PPNT    002430            534*    703     705*    706     708*    735*
PRESTR  006624           1992*   2142
PRIME   001234            487     529*
PSW     001232            472*    488*    527*
PTBCOT  007400           2335*
PUNCHR  002434            704*    710     737*
```

```
PWRFAL= 000024  2571*  2592   2609   2897
PWRFL   001674   446    601*   605
PWRUP   001704   601    604*
P10IB = 025612  2559*  2758
P10IC = 045772  2564*  2565   2707   2758*  2759*
P10IS = 025610  2556*  2559   2598*  2756   2760*
P100B = 025616  2560*  2742*  2753*  2861*
P100C = 045776  2563*  2564   2738   2742   2743*  2751   2859
P100S = 025614  2555*  2560   2600*  2740
RCVEC   007764  2481*
RC11    007720  2462*  2481
REDCHR  002436   723*   724*   725    738*
RELATV= 130000   187*  1200   1212   2208*  2292
RES     007506  2373*  2388
RESTRT  006060  1746*  1992
RFVEC   007714  2459*
RF11    007600  2410*  2459
RKVEC   007760  2479*
RK11    007610  2415*  2479
ROMADD  001014   467*   799
ROMORG= 166000  2547*
RCTVAL  002052   643*   660
RPNT    002432   535*   722    726*   727    729*   730    736*
RPVEC   007770  2483*
RP11    007654  2439*  2483
RSTRT   016034  2604*  2721   2808
SENDIT  016730  2807   2836   2858*  2860   2864
SETDUN  006126  1761   1770*
SETLP1  006074  1752*  1754
SETLP2  006110  1760*  1767
SETLP3  006116  1764*  1766
SETUP   006432  1757   1914*
SHORTV= 104000   187*  1224   1236   2203*  2280   2282   2284   2286
SIZE    001312   539*   624*   625*   628
START   006000   480   1710*
STARTA  016000   483   2592*
STARTB  001022   456    471*   606    674    692    699    719    747    836
STARTX= 000000  2549*  2893
STARTY= 001360  2550*  2894
STATSA= 170000   187*  1290   1305   1322   1340   1358   1376   1394   2231*
STATSB= 174000   187*   581    680    682    846    874   2241*  2301
STKSRT= 015770  2565*  2604   2783
STOP    007550  2387*  2391
SWR     001230   475    477    526*   564*   565    569*
SWREG   000170   452*   569
SYNOFF= 000010   187*
SYNON = 000004   187*  2238*
TAB     006222  1805   1823*  1827
TABLE   007562  2374   2394*
TABOOT  007500  2371*  2401
TACS  = 177500  2369*  2371
TAPES   007662  2422   2431   2444*
TCVEC   007774  2485*
TC11    007620  2421*  2485
TKB     001214   517*   752
TKS     001212   516*   532*   744
```

```
TMEM      002702       798     815#
TMPEND=   007776       1658#   1712    1746
TM11      007636       2430#
TPB       001220       519#    819#    822#
TPS       001216       518#    620     823
VT        006244       1807    1831#
VT40PC=   022000       2553#   2612#
WCFDS     001016       469#    479#    482#    809
XMLOP1    002020       632#    637
XMLOP2    002024       633#    635
XMRTS     002040       631     638#
.       = 017002       188#    191     193     195     197     199     201     203     205     207     209     211     213
                       215     217     219     221     223     225     227     229     231     233     235     237     239
                       241     243     245     247     249     251     253     255     257     259     261     263     265
                       267     269     271     273     275     277     279     281     283     285     297     289     291
                       293     295     297     299     301     303     305     307     309     311     313     315     317
                       319     321     323     325     327     329     331     333     335     337     329     341     343
                       345     347     349     351     353     355     357     359     361     363     365     367     369
                       371     373     375     377     379     381     383     385     387     369     391     393     395
                       397     399     401     403     405     407     409     411     413     415     417     419     421
                       423     425     427     429     431     433     435     437     439     441     443     445#    451#
                       454#    458#    653     655     658     661     697     717     745     1696#   2352    2374    2452
                       2474    2583#   2652    2688    2838
```

. ABS. 017002 000

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

DDGTEC,BIN,DDGTEC,SEQ/SQ:/CRF NL:TOC=DDGTEC.SML,DDGTEC.P11
RUN-TIME: 24 32 1 SECONDS
RUN-TIME RATIO: 123/59=2.0
CORE USED: 33K (65 PAGES)

Spooler routine 0 Seconds, 39 KCS, 210 disk reads, 3 disk writes, 67 pages

[redacted]