

!A

\*\*\* End of Pass 1

\*\*\* End of Pass 2

```
0800      1          ttl "Insta-Disk DATALOAD80, DATALOAD80.L"
0800      2      ;
0800      3      ;
0800      4      ; DATALOAD80.L
0800      5      ;
0800      6      ; 2024 February 14
0800      7      ;
0800      8      ;
0800      9      ; This softawre is based on the work of Egan Ford.
0800     10      ;
0800     11      ;
0800     12      ; DOS 4.5, Build 06
0800     13      ;
0800     14      ; 2024 February 14
0800     15      ;
0800     16      ;
0800     17      ; Start of Source Code: 0x4000
0800     18      ; Start of Symbol List: 0x7800
0800     19      ;
0800     20      ;
0800     21      ; Copyright (c) 2024 February 14 by
0800     22      ; Walland Philip Vrbancic Jr.
0800     23      ;
0800     24      ; 6223 East Peabody Street
0800     25      ; Long Beach, California 90808
0800     26      ; United States of America
0800     27      ;
0800     28      ; All Rights Reserved
0800     29      ;
0800     30      ; This software is the confidential and
0800     31      ; proprietary intellectual property of
0800     32      ; Walland Philip Vrbancic Jr.
0800     33      ;
0800     34      ;
0000     35  DATAPTR    epz $00
0002     36  ENDPTR    epz $02
0004     37  CHKSUM    epz $04
0008     38  SRCPTR    epz $08
000A     39  DSTPTR    epz $0A
000F     40  PAGEBYTE  epz $0F
0800     41      ;
0800     42          enz
0800     43      ;
0000     44  ZERO      equ $00
00FF     45  NEGONE    equ $FF
0800     46      ;
000F     47  NIBLMASK  equ $0F
0800     48      ;
008D     49  RETURN    equ $8D
0800     50      ;
0006     51  BITVAL    equ $06
000D     52  HDRVAL    equ $0D
0038     53  ENDVAL    equ $38
0800     54      ;
0100     55  PAGESIZE  equ $100
0800     56      ;
04FB     57  XMODE     equ $4FB
0800     58      ;
07D0     59  LINE23    equ $7D0
0800     60      ;
```

```

0810      61  DISKLOAD equ $810
0800      62  ;
2000      63  DISKADR equ $2000
0600      64  DISKLEN equ $600
0800      65  ;
0200      66  INFLLEN equ $200
0800      67  ;
1A00      68  DOSLEN1 equ $1A00
0E00      69  DOSLEN2 equ $E00
0200      70  DOSLEN3 equ $200
2A00      71  DOSLEN equ DOSLEN1+DOSLEN2+DOSLEN3
0800      72  ;
B000      73  DATALOAD equ $B000
B200      74  DISKCOPY equ $B200
B800      75  INFLATE equ $B800
0800      76  ;
D000      77  DOSADR1 equ $D000
D000      78  DOSADR2 equ $D000
BE00      79  DOSADR3 equ $BE00
BFF6      80  MNGUSER equ $BFF6
BFF8      81  INITDOS equ $BFF8
0800      82  ;
C00C      83  VID80OFF equ $C00C
C00E      84  ALTCHOFF equ $C00E
0800      85  ;
C010      86  CONNECT equ $C010
0800      87  ;
C060      88  TAPEIN equ $C060
0800      89  ;
C082      90  ROM2WP equ $C082
C083      91  RAM2WE equ $C083
C08B      92  RAM1WE equ $C08B
0800      93  ;
F941      94  PRNTAX equ $F941
FB2F      95  INIT equ $FB2F
FDDA      96  PRBYTE equ $FDDA
FDED      97  COUT equ $FDED
FE84      98  SETNORM equ $FE84
FE89      99  SETKBD equ $FE89
FE93     100  SETVID equ $FE93
0800     101  ;
FF69     102  MONITOR equ $FF69
0800     103  ;
0800     104  ;
0810     105          org DISKLOAD
0810     106          obj DISKLOAD
0810     107          usr
0810     108  ;
0810     109  ;
0810     110  ; Take control of the keyboard and video and set up for a
0810     111  ; normal display.
0810     112  ;
0810 A2 FF     113          ldx #NEGONE
0812 9A       114          txs
0813         115  ;
0813 8E FB 04  116          stx XMODE
0816 8E 0C C0  117          stx VID80OFF
0819 8E 0E C0  118          stx ALTCHOFF
081C         119  ;
081C 20 84 FE  120          jsr SETNORM
081F 20 2F FB  121          jsr INIT

```

```

0822 20 93 FE    122          jsr SETVID
0825 20 89 FE    123          jsr SETKBD
0828            124          ;
0828            125          ;
0828            126          ; Print TEXT1, copy DATALOAD80 to DATALOAD at 0xB000,
0828            127          ; and call DATALOAD as a subroutine. Up to 0x3200 bytes
0828            128          ; are read and transferred to 0x2000. DISKCOPY is found
0828            129          ; at 0x2000 and its 0x600 bytes are copied to 0xB200. If
0828            130          ; CMPLOAD is not ZERO, INFLATE is found next and its 0x200
0828            131          ; bytes are copied to 0xB800. If DOSLOAD is not ZERO,
0828            132          ; DOS 4.5.06H is found next and its 0x2A00 bytes are copied
0828            133          ; to the Language Card, DOS is initialized, and the Page
0828            134          ; 0x03 Vectors are created. A running checksum is
0828            135          ; calculated in this version of DATALOAD80.
0828            136          ;
0828 A2 03        137          ldx #3
082A            138          ;
082A BD 5F 0A    139 ^1      lda LOADTIME,X
082D 9D 14 09    140          sta TEXTMOD,X
0830            141          ;
0830 CA          142          dex
0831 10 F7        143          bpl <1
0833            144          ;
0833 A0 00        145          ldy #TEXT1-TEXTS
0835 20 EE 08    146          jsr PRNTTEXT
0838            147          ;
0838            148          ;
0838            149          ; Install DATALOAD at 0xB000 and read up to 0x3200 bytes.
0838            150          ;
0838 A2 00        151          ldx #ZERO
083A            152          ;
083A BD 23 09    153 ^2      lda DATACODE,X
083D 9D 00 B0    154          sta DATALOAD,X
0840            155          ;
0840 E8          156          inx
0841 D0 F7        157          bne <2
0843            158          ;
0843 A9 01        159          lda /CODELEN
0845 F0 0B        160          beq >4
0847            161          ;
0847 BD 23 0A    162 ^3      lda DATACODE+PAGESIZE,X
084A 9D 00 B1    163          sta DATALOAD+PAGESIZE,X
084D            164          ;
084D E8          165          inx
084E            166          ;
084E E0 3C        167          cpx #CODELEN
0850 D0 F5        168          bne <3
0852            169          ;
0852 A0 00        170 ^4      ldy #DISKADR
0854 A9 20        171          lda /DISKADR
0856            172          ;
0856 84 00        173          sty DATAPTR
0858 85 01        174          sta DATAPTR+1
085A            175          ;
085A 84 08        176          sty SRCPTR
085C 85 09        177          sta SRCPTR+1
085E            178          ;
085E 84 0A        179          sty DSTPTR
0860            180          ;
0860 18          181          clc
0861            182          ;

```

```

0861 69 06      183      adc /DISKLEN
0863           184      ;
0863 AE 63 0A    185      ldx CMPLOAD
0866 F0 02      186      beq >5
0868           187      ;
0868 69 02      188      adc /INFLLEN
086A           189      ;
086A AE 64 0A    190      ^5 ldx DOSLOAD
086D F0 02      191      beq >6
086F           192      ;
086F 69 2A      193      adc /DOSLEN
0871           194      ;
0871 C8          195      ^6 iny                ; modified EOF handler
0872           196      ;
0872 84 02      197      sty ENDPTR
0874 85 03      198      sta ENDPTR+1
0876           199      ;
0876 20 00 B0    200      jsr DATALOAD
0879           201      ;
0879           202      ;
0879           203      ; Move DISKCOPY to 0xB200.
0879           204      ;
0879 A0 00      205      ldy #ZERO
087B           206      ;
087B A9 B2      207      lda /DISKCOPY
087D A2 06      208      ldx /DISKLEN
087F           209      ;
087F 20 D7 08    210      jsr COPYPGS
0882           211      ;
0882           212      ;
0882           213      ; Move INFLATE to 0xB800 if CMPLOAD is not ZERO.
0882           214      ;
0882 AD 63 0A    215      lda CMPLOAD
0885 F0 07      216      beq >7
0887           217      ;
0887 A9 B8      218      lda /INFLATE
0889 A2 02      219      ldx /INFLLEN
088B           220      ;
088B 20 D7 08    221      jsr COPYPGS
088E           222      ;
088E           223      ;
088E           224      ; Move DOS 4.5.06H into the Language Card if DOSLOAD is not
088E           225      ; ZERO.
088E           226      ;
088E AD 64 0A    227      ^7 lda DOSLOAD
0891 F0 33      228      beq >8
0893           229      ;
0893 2C 83 C0    230      bit RAM2WE
0896 2C 83 C0    231      bit RAM2WE
0899           232      ;
0899 A9 D0      233      lda /DOSADR1
089B A2 1A      234      ldx /DOSLEN1
089D           235      ;
089D 20 D7 08    236      jsr COPYPGS
08A0           237      ;
08A0 2C 8B C0    238      bit RAM1WE
08A3 2C 8B C0    239      bit RAM1WE
08A6           240      ;
08A6 A9 D0      241      lda /DOSADR2
08A8 A2 0E      242      ldx /DOSLEN2
08AA           243      ;

```

```

08AA 20 D7 08      244      jsr COPYPGS
08AD              245      ;
08AD 2C 82 C0      246      bit ROM2WP
08B0              247      ;
08B0 A9 BE          248      lda /DOSADR3
08B2 A2 02          249      ldx /DOSLEN3
08B4              250      ;
08B4 20 D7 08      251      jsr COPYPGS
08B7              252      ;
08B7              253      ;
08B7              254      ; Setup MNGUSR to return to RTNUSER after DOS 4.5.06H has
08B7              255      ; fully initialized and has created the PAGE3 Vectors.
08B7              256      ;
08B7 38            257      sec
08B8              258      ;
08B8 A0 C2          259      ldy #RTNUSER
08BA A9 08          260      lda /RTNUSER
08BC              261      ;
08BC 20 E8 08      262      jsr SETUSER
08BF              263      ;
08BF 6C F8 BF      264      jmp (INITDOS)
08C2              265      ;
08C2              266      ;
08C2              267      ; DOS 4.5.06H has initialized and the Page 0x03 Vectors are
08C2              268      ; now available for the RWTS routines. Return MNGUSR to
08C2              269      ; its default state and enter DISKCOPY.
08C2              270      ;
08C2 18            271      RTNUSER clc
08C3              272      ;
08C3 20 E8 08      273      jsr SETUSER
08C6              274      ;
08C6              275      ;
08C6              276      ; If SCNTFLAG is not ZERO, connect DOS to the disk device
08C6              277      ; located at the SLOT CX address.
08C6              278      ;
08C6 AD 65 0A      279      ^8      lda SCNTFLAG
08C9 F0 09          280      beq >9
08CB              281      ;
08CB AD 66 0A      282      lda SLOT CX
08CE 8D ED 08      283      sta SLOT CNCT+2
08D1              284      ;
08D1 20 EB 08      285      jsr SLOT CNCT
08D4              286      ;
08D4              287      ;
08D4              288      ; Ready to begin DISKCOPY.
08D4              289      ;
08D4 4C 00 B2      290      ^9      jmp DISKCOPY
08D7              291      ;
08D7              292      ;
08D7              293      ; SRCPTR needs to be initialized once. All routines are
08D7              294      ; copied sequentially to where they need to reside.
08D7              295      ;
08D7 85 0B          296      COPYPGS sta DSTPTR+1
08D9              297      ;
08D9 B1 08          298      ^1      lda (SRCPTR),Y
08DB 91 0A          299      sta (DSTPTR),Y
08DD              300      ;
08DD C8            301      iny
08DE D0 F9          302      bne <1
08E0              303      ;
08E0 E6 09          304      inc SRCPTR+1

```

```

08E2 E6 0B      305      inc DSTPTR+1
08E4           306      ;
08E4 CA        307      dex
08E5 D0 F2     308      bne <1
08E7           309      ;
08E7 60        310      rts
08E8           311      ;
08E8           312      ;
08E8 6C F6 BF  313      SETUSER  jmp (MNGUSER)
08EB           314      ;
08EB 4C 10 C0  315      SLOTCNCT jmp CONNECT
08EE           316      ;
08EE           317      ;
08EE B9 FA 08  318      PRNTTEXT lda TEXTS,Y
08F1 F0 06     319      beq >1
08F3           320      ;
08F3 20 ED FD  321      jsr COUT
08F6           322      ;
08F6 C8        323      iny
08F7 D0 F5     324      bne PRNTTEXT      ; always taken
08F9           325      ;
08F9 60        326      ^1      rts
08FA           327      ;
08FA           328      ;
08FA           329      TEXTS:
08FA           330      ;
08FA 8D        331      TEXT1    byt RETURN
08FB C9 EE F3  332      asc "Installing c2t routines, "
08FE F4 E1 EC
0901 EC E9 EE
0904 E7 A0 E3
0907 B2 F4 A0
090A F2 EF F5
090D F4 E9 EE
0910 E5 F3 AC
0913 A0
0914 B0 B0 AE  333      TEXTMOD  asc "00.0 seconds"
0917 B0 A0 F3
091A E5 E3 EF
091D EE E4 F3
0920 8D 8D 00  334      byt RETURN,RETURN,ZERO
0923           335      ;
0923           336      ;
0923           337      ; I developed this version of the DATALOAD code using ideas
0923           338      ; from the DISKLOAD8000 and the DISKLOAD9600 routines. The
0923           339      ; c2t software should adjust the second half of the 1's
0923           340      ; waveform to match the second half of the 0's waveform.
0923           341      ; Testing has verified that this routine has no timing
0923           342      ; issues.
0923           343      ;
0923           344      DATACODE:
0923           345      ;
0923           346      phs DATALOAD
B000           347      ;
B000           348      CODEBGN:
B000           349      ;
B000           350      ; Initialize the DATALOAD routine.
B000           351      ;
B000 A9 FF     352      lda #NEGONE
B002 85 04     353      sta CHKSUM
B004           354      ;

```

```

B004 A0 00      355      ldy #ZERO
B006 8C FE B0   356      sty VAL1
B009 8C FF B0   357      sty VAL2
B00C 8C 00 B1   358      sty VAL3
B00F           359      ;
B00F           360      ;
B00F           361      ; Consume two of the first waveforms found.
B00F           362      ;
B00F 2C 60 C0   363      ^0      bit TAPEIN
B012 30 FB      364      bmi <0
B014           365      ;
B014 2C 60 C0   366      ^1      bit TAPEIN
B017 10 FB      367      bpl <1
B019           368      ;
B019           369      ;
B019           370      ; Display a running counter while receiving HDR data.
B019           371      ; Initialize the beginning of a new data byte.
B019           372      ;
B019 20 C0 B0   373      ^2      jsr SHOWVAL
B01C           374      ;
B01C A9 01      375      lda #1
B01E           376      ;
B01E           377      ;
B01E           378      ; Initialize the read bit loop.
B01E           379      ;
B01E A2 00      380      ^3      ldx #ZERO
B020           381      ;
B020           382      ;
B020           383      ; Consume the negative half of the old waveform.
B020           384      ;
B020 2C 60 C0   385      ^4      bit TAPEIN
B023 30 FB      386      bmi <4
B025           387      ;
B025           388      ;
B025           389      ; Measure the positive half of the new waveform.
B025           390      ;
B025 E8         391      ^5      inx
B026           392      ;
B026 2C 60 C0   393      bit TAPEIN
B029 10 FA      394      bpl <5
B02B           395      ;
B02B           396      ;
B02B           397      ; Test measurement for END, HDR, or BIT.
B02B           398      ;
B02B E0 38      399      cpx #ENDVAL
B02D B0 1F      400      bcs >7
B02F           401      ;
B02F E0 0D      402      cpx #HDRVAL
B031 B0 E6      403      bcs <2
B033           404      ;
B033 E0 06      405      cpx #BITVAL
B035           406      ;
B035           407      ;
B035           408      ; Capture bit value. Test for data byte complete.
B035           409      ;
B035 2A         410      rol
B036 90 E6      411      bcc <3
B038           412      ;
B038           413      ;
B038           414      ; Save data byte and update checksum.
B038           415      ;

```

```

B038 91 00      416      sta (DATAPTR),Y
B03A           417      ;
B03A 45 04      418      eor CHKSUM
B03C 85 04      419      sta CHKSUM
B03E           420      ;
B03E           421      ;
B03E           422      ; Consume negative half of this waveform.
B03E           423      ;
B03E 2C 60 C0   424      ^6      bit TAPEIN
B041 30 FB      425      bmi <6
B043           426      ;
B043           427      ;
B043           428      ; Initialize registers and test for page complete.
B043           429      ;
B043 A9 01      430      lda #1
B045 AA         431      tax
B046           432      ;
B046 C8         433      iny
B047 D0 DC      434      bne <5
B049           435      ;
B049           436      ;
B049           437      ; Point to next page and continue.
B049           438      ;
B049 E8         439      inx
B04A           440      ;
B04A E6 01      441      inc DATAPTR+1
B04C D0 D7      442      bne <5                      ; always taken
B04E           443      ;
B04E           444      ;
B04E           445      ; EOF handler. The 770 Hz Header signal has been detected.
B04E           446      ; Add the Y-reg to the DATAPTR.
B04E           447      ;
B04E 18         448      ^7      clc
B04F           449      ;
B04F 98         450      tya
B050 85 0F      451      sta PAGEBYTE                ; save Y-reg for analysis
B052           452      ;
B052 65 00      453      adc DATAPTR
B054 85 00      454      sta DATAPTR
B056           455      ;
B056 A5 01      456      lda DATAPTR+1
B058 69 00      457      adc #ZERO
B05A 85 01      458      sta DATAPTR+1
B05C           459      ;
B05C           460      ;
B05C           461      ; Print beginning of end address status message.
B05C           462      ;
B05C A0 00      463      ldy #MSG1-MESGS          ; End address is
B05E 20 B4 B0   464      jsr PRNTEMESG
B061           465      ;
B061           466      ;
B061           467      ; Test end address for correctness.
B061           468      ;
B061 A5 00      469      lda DATAPTR
B063 C5 02      470      cmp ENDPTR
B065 D0 2B      471      bne >8
B067           472      ;
B067 A5 01      473      lda DATAPTR+1
B069 C5 03      474      cmp ENDPTR+1
B06B D0 25      475      bne >8
B06D           476      ;

```

```

B06D      477 ;
B06D      478 ; Complete address status and begin CHKSUM status message.
B06D      479 ;
B06D A0 23 480      ldy #MSG3-MESGS      ; Okay.
B06F 20 B4 B0 481      jsr PRNMSG
B072      482 ;
B072 A0 11 483      ldy #MSG2-MESGS      ; CHKSUM value is
B074 20 B4 B0 484      jsr PRNMSG
B077      485 ;
B077      486 ;
B077      487 ; Test CHKSUM for correctness.
B077      488 ;
B077 A5 04 489      lda CHKSUM
B079 F0 37 490      beq >9
B07B      491 ;
B07B      492 ;
B07B      493 ; Complete CHKSUM error message.
B07B      494 ;
B07B A0 29 495      ldy #MSG4-MESGS      ; 0x
B07D 20 B4 B0 496      jsr PRNMSG
B080      497 ;
B080 A5 04 498      lda CHKSUM
B082 20 DA FD 499      jsr PRBYTE
B085      500 ;
B085 A0 2C 501      ldy #MSG5-MESGS      ; and not 0x
B087 20 B4 B0 502      jsr PRNMSG
B08A      503 ;
B08A A9 00 504      lda #ZERO
B08C 20 DA FD 505      jsr PRBYTE
B08F      506 ;
B08F 4C AA B0 507      jmp ERROR
B092      508 ;
B092      509 ;
B092      510 ; Complete end address error message.
B092      511 ;
B092 A0 29 512      ^8 ldy #MSG4-MESGS      ; 0x
B094 20 B4 B0 513      jsr PRNMSG
B097      514 ;
B097 A6 00 515      ldx DATAPTR
B099 A5 01 516      lda DATAPTR+1
B09B      517 ;
B09B 20 41 F9 518      jsr PRNTAX
B09E      519 ;
B09E A0 2C 520      ldy #MSG5-MESGS      ; and not 0x
B0A0 20 B4 B0 521      jsr PRNMSG
B0A3      522 ;
B0A3 A6 02 523      ldx ENDPTR
B0A5 A5 03 524      lda ENDPTR+1
B0A7      525 ;
B0A7 20 41 F9 526      jsr PRNTAX
B0AA      527 ;
B0AA      528 ;
B0AA      529 ; Enter the Monitor because an error occurred.
B0AA      530 ;
B0AA A0 38 531      ERROR ldy #MSG6-MESGS      ; .
B0AC 20 B4 B0 532      jsr PRNMSG
B0AF      533 ;
B0AF 4C 69 FF 534      jmp MONITOR
B0B2      535 ;
B0B2      536 ;
B0B2      537 ; Complete CHKSUM status and return to caller.

```

```

B0B2          538 ;
B0B2 A0 23    539 ^9      ldy #MSG3-MESGS      ; Okay.
B0B4          540 ;
B0B4          541 ;
B0B4 B9 01 B1 542 PRNTMSG lda MSGS,Y
B0B7 F0 06    543          beq >1
B0B9          544 ;
B0B9 20 ED FD 545          jsr COUT
B0BC          546 ;
B0BC C8       547          iny
B0BD D0 F5    548          bne PRNTMSG      ; always taken
B0BF          549 ;
B0BF 60       550 ^1      rts
B0C0          551 ;
B0C0          552 ;
B0C0          553 ; Increment the HDR counter in BCD.
B0C0          554 ;
B0C0 F8       555 SHOWVAL sed
B0C1          556 ;
B0C1 18       557          clc
B0C2          558 ;
B0C2 AD 00 B1 559          lda VAL3
B0C5 69 01    560          adc #1
B0C7 8D 00 B1 561          sta VAL3
B0CA          562 ;
B0CA AD FF B0 563          lda VAL2
B0CD 69 00    564          adc #ZERO
B0CF 8D FF B0 565          sta VAL2
B0D2          566 ;
B0D2 AD FE B0 567          lda VAL1
B0D5 69 00    568          adc #ZERO
B0D7 8D FE B0 569          sta VAL1
B0DA          570 ;
B0DA D8       571          cld
B0DB          572 ;
B0DB A2 22    573          ldx #$22
B0DD          574 ;
B0DD 20 E9 B0 575          jsr SHOWBYTE
B0E0          576 ;
B0E0 AD FF B0 577          lda VAL2
B0E3 20 E9 B0 578          jsr SHOWBYTE
B0E6          579 ;
B0E6 AD 00 B1 580          lda VAL3
B0E9          581 ;
B0E9          582 ;
B0E9          583 ; Display a counter byte.
B0E9          584 ;
B0E9 48       585 SHOWBYTE pha
B0EA          586 ;
B0EA 4A       587          lsr
B0EB 4A       588          lsr
B0EC 4A       589          lsr
B0ED 4A       590          lsr
B0EE          591 ;
B0EE 09 B0    592          ora #"0"
B0F0 9D D0 07 593          sta LINE23,X
B0F3          594 ;
B0F3 E8       595          inx
B0F4          596 ;
B0F4 68       597          pla
B0F5 29 0F    598          and #NIBLMASK

```

```

B0F7          599 ;
B0F7 09 B0    600      ora #"0"
B0F9 9D D0 07 601      sta LINE23,X
B0FC          602 ;
B0FC E8       603      inx
B0FD          604 ;
B0FD 60       605      rts
B0FE          606 ;
B0FE          607 ;
B0FE          608 VAL1      dfs 1,ZERO
B0FF          609 VAL2      dfs 1,ZERO
B100          610 VAL3      dfs 1,ZERO
B101          611 ;
B101          612 ;
B101          613 MSGS:
B101          614 ;
B101 8D       615 MSG1      byt RETURN
B102 C5 EE E4 616      asc "End address is "
B105 A0 E1 E4
B108 E4 F2 E5
B10B F3 F3 A0
B10E E9 F3 A0
B111 00       617      byt ZERO
B112          618 ;
B112 8D       619 MSG2      byt RETURN
B113 C3 C8 CB 620      asc "CHKSUM value is "
B116 D3 D5 CD
B119 A0 F6 E1
B11C EC F5 E5
B11F A0 E9 F3
B122 A0
B123 00       621      byt ZERO
B124          622 ;
B124 CF EB E1 623 MSG3      asc "Okay."
B127 F9 AE
B129 00       624      byt ZERO
B12A          625 ;
B12A B0 F8    626 MSG4      asc "0x"
B12C 00       627      byt ZERO
B12D          628 ;
B12D A0 E1 EE 629 MSG5      asc " and not 0x"
B130 E4 A0 EE
B133 EF F4 A0
B136 B0 F8
B138 00       630      byt ZERO
B139          631 ;
B139 AE       632 MSG6      asc "."
B13A 8D 00    633      byt RETURN,ZERO
B13C          634 ;
B13C          635 ;
013C          636 CODELEN equ *-CODEBGN
B13C          637 ;
B13C          638 ;
B13C          639      phs DATACODE+CODELEN
0A5F          640 ;
0A5F          641 ;
0A5F          642 ; The following values are supplied by c2t.
0A5F          643 ;
0A5F B0 B0 AE 644 LOADTIME asc "00.0" ; time to install all routines
0A62 B0
0A63          645 ;

```

0A63 00	646	CMPLOAD	hex 00	; compress load flag
0A64 00	647	DOSLOAD	hex 00	; DOS load flag
0A65	648			
0A65 00	649	SCNTFLAG	hex 00	; slot connect flag
0A66 00	650	SLOT CX	hex 00	; slot CX MSB address
0A67	651			
0A67	652			

BSAVE DATALOAD80,A\$0810,B,L\$0257

0A67	653	usr DATALOAD80
0A67	654	;
0A67	655	;
0A67	656	stt "DATALOAD80 Symbol Table"
0A67	657	;
0A67	658	;
0A67	659	end 111

\*\*\* End of Assembly

Symbol List starts at 0x7800, ends at 0x7B16, used 0x0316, remaining 0x3C42

### Symbols unsorted:

DATAPTR	0000	ENDPTR	0002	CHKSUM	0004	SRCPTR	0008	DSTPTR	000A
PAGEBYTE	000F	ZERO	0000	NEGONE	00FF	NIBLMASK	000F	RETURN	008D
BITVAL	0006	HDRVAL	000D	ENDVAL	0038	PAGESIZE	0100	XMODE	04FB
LINE23	07D0	DISKLOAD	0810	DISKADR	2000	DISKLEN	0600	INFLLEN	0200
DOSLEN1	1A00	DOSLEN2	0E00	DOSLEN3	0200	DOSLEN	2A00	DATALOAD	B000
DISKCOPY	B200	INFLATE	B800	DOSADR1	D000	DOSADR2	D000	DOSADR3	BE00
MNGUSER	BFF6	INITDOS	BFF8	VID80OFF	C00C	ALTCHOFF	C00E	CONNECT	C010
TAPEIN	C060	ROM2WP	C082	RAM2WE	C083	RAM1WE	C08B	PRNTAX	F941
INIT	FB2F	PRBYTE	FDDA	COUT	FDED	SETNORM	FE84	SETKBD	FE89
SETVID	FE93	MONITOR	FF69	RTNUSER	08C2	COPYPGS	08D7	SETUSER	08E8
SLOTNCNT	08EB	PRNTTEXT	08EE	TEXTS	08FA	TEXT1	08FA	TEXTMOD	0914
DATACODE	0923	CODEBGN	B000	ERROR	B0AA	PRNTMSG	B0B4	SHOWVAL	B0C0
SHOWBYTE	B0E9	VAL1	B0FE	VAL2	B0FF	VAL3	B100	MESGS	B101
MESG1	B101	MESG2	B112	MESG3	B124	MESG4	B12A	MESG5	B12D
MESG6	B139	CODELEN	013C	LOADTIME	0A5F	CMPLOAD	0A63	DOSLOAD	0A64
SCNTFLAG	0A65	SLOTX	0A66						

### Symbols alphabetically sorted:

ALTCHOFF	C00E	BITVAL	0006	CHKSUM	0004	CMPLOAD	0A63	CODEBGN	B000
CODELEN	013C	CONNECT	C010	COPYPGS	08D7	COUT	FDED	DATACODE	0923
DATALOAD	B000	DATAPTR	0000	DISKADR	2000	DISKCOPY	B200	DISKLEN	0600
DISKLOAD	0810	DOSADR1	D000	DOSADR2	D000	DOSADR3	BE00	DOSLEN	2A00
DOSLEN1	1A00	DOSLEN2	0E00	DOSLEN3	0200	DOSLOAD	0A64	DSTPTR	000A
ENDPTR	0002	ENDVAL	0038	ERROR	B0AA	HDRVAL	000D	INFLATE	B800
INFLLEN	0200	INIT	FB2F	INITDOS	BFF8	LINE23	07D0	LOADTIME	0A5F
MESG1	B101	MESG2	B112	MESG3	B124	MESG4	B12A	MESG5	B12D
MESG6	B139	MESGS	B101	MNGUSER	BFF6	MONITOR	FF69	NEGONE	00FF
NIBLMASK	000F	PAGEBYTE	000F	PAGESIZE	0100	PRBYTE	FDDA	PRNTAX	F941
PRNTMSG	B0B4	PRNTTEXT	08EE	RAM1WE	C08B	RAM2WE	C083	RETURN	008D
ROM2WP	C082	RTNUSER	08C2	SCNTFLAG	0A65	SETKBD	FE89	SETNORM	FE84
SETUSER	08E8	SETVID	FE93	SHOWBYTE	B0E9	SHOWVAL	B0C0	SLOTNCNT	08EB
SLOTX	0A66	SRCPTR	0008	TAPEIN	C060	TEXT1	08FA	TEXTMOD	0914
TEXTS	08FA	VAL1	B0FE	VAL2	B0FF	VAL3	B100	VID80OFF	C00C
XMODE	04FB	ZERO	0000						

### Symbols numerically sorted:

ZERO	0000	DATAPTR	0000	ENDPTR	0002	CHKSUM	0004	BITVAL	0006
SRCPTR	0008	DSTPTR	000A	HDRVAL	000D	PAGEBYTE	000F	NIBLMASK	000F
ENDVAL	0038	RETURN	008D	NEGONE	00FF	PAGESIZE	0100	CODELEN	013C
INFLLEN	0200	DOSLEN3	0200	XMODE	04FB	DISKLEN	0600	LINE23	07D0
DISKLOAD	0810	RTNUSER	08C2	COPYPGS	08D7	SETUSER	08E8	SLOTNCNT	08EB
PRNTTEXT	08EE	TEXTS	08FA	TEXT1	08FA	TEXTMOD	0914	DATACODE	0923
LOADTIME	0A5F	CMPLOAD	0A63	DOSLOAD	0A64	SCNTFLAG	0A65	SLOTX	0A66
DOSLEN2	0E00	DOSLEN1	1A00	DISKADR	2000	DOSLEN	2A00	DATALOAD	B000
CODEBGN	B000	ERROR	B0AA	PRNTMSG	B0B4	SHOWVAL	B0C0	SHOWBYTE	B0E9
VAL1	B0FE	VAL2	B0FF	VAL3	B100	MESGS	B101	MESG1	B101
MESG2	B112	MESG3	B124	MESG4	B12A	MESG5	B12D	MESG6	B139
DISKCOPY	B200	INFLATE	B800	DOSADR3	BE00	MNGUSER	BFF6	INITDOS	BFF8
VID80OFF	C00C	ALTCHOFF	C00E	CONNECT	C010	TAPEIN	C060	ROM2WP	C082
RAM2WE	C083	RAM1WE	C08B	DOSADR2	D000	DOSADR1	D000	PRNTAX	F941
INIT	FB2F	PRBYTE	FDDA	COUT	FDED	SETNORM	FE84	SETKBD	FE89
SETVID	FE93	MONITOR	FF69						