


```
0000 1: ;-----
0000 2: ;      Copyright 2012 Integrated Logic Systems
0000 3: ;      Source Code is Copyright Stephen J. Car
0000 4: ;
0000 5: ;
0000 6: ;      Please do not share this source!
0000 7: ;-----
0000 8: ;      SetUTime.com
0000 9: ;
0000 10: ;This is a program gets or sets the time clock in the sio2
0000 11: ;      init  9/09/2012    sjc new design
0000 12: ;
0000 13: ;
0000 14: ;
0000 15: ;-----
0000 16: ; Notes: o This source code MAY NOT be placed for download
0000 17: ;      o "mea" is a macro that loads the address of the
0000 18: ;      into the pointer specified by the second field.
0000 19: ;-----
0000 20: ; Assembler: MADMAC (tm) ST Cross Assembler (Atari Corp)
0000 21: ;      XASM  st and IBM VERSIONS
0000 22: ;-----
0000 23:
0000 24: com_file_name:      .macro
0000 25: ;      dc.b  13,"filename.ext",$9b
0000 26: ;      dc.b  12,"SetUTime.com",$9b
0000 27: ;      .endm
0000 28:
0000 29: ;
0000 30: ;-----
0000 31: ; File revision history. Version Number in hex
0000 32: ;
0000 33: file_ver:      equ  $10
0000 34: ;-----
0000 35: ; the Month in dec for the first time this revision
0000 36: ;was compiled
0000 37: ;
0000 38: c_month:      equ  9
0000 39: ;-----
0000 40: ; the day in dec for the first time this revision
0000 41: ;was compiled
0000 42: c_day:      equ  16
0000 43: ;-----
0000 44: ; the year in dec for the first time this revision
0000 45: ;was compiled
0000 46: c_year:      equ  2012
0000 47: ;-----
0000 48: ; Type of compiler used in program
0000 49: ;
0000 50: ; 0 = unknown Compiler
0000 51: ; 1 = xasm
0000 52: ; 2 = mac_65
0000 53: ; 3 = basic
```

```
0000      54: ; 4 = compiled basic xl
0000      55: ; 5 = C65
0000      56: ; 6 = Action
0000      57: ;
0000      58: ; We can add more as time goes on!
0000      59: ;
0000      60: ;
0000      61: ;
0000      62: xasm:          equ    1
0000      63: mac_65:       equ    2
0000      64: basic:       equ    3
0000      65: basicxl:     equ    4
0000      66: c65:        equ    5
0000      67: action:      equ    6
0000      68: ;
0000      69: file_compiler:    equ    xasm
0000      70: ;-----
0000      71: ; is this relocatable code ?
0000      72: ; anyother value other than 1 or 2 would be unknown
0000      73: r..yes:          equ    1
0000      74: r..no:         equ    2
0000      75: ;
0000      76: ;-----
0000      77: ; Gotta define if it can be relocated
0000      78: l_relocatable:    equ    r..no
0000      79: ;
0000      80: ;
0000      81: ;-----
0000      82: ; Gotta define if it can be relocated
0000      83: r..crl:          equ    $7d
0000      84: r..crlf:         equ    $9b
0000      85: r..space:        equ    $20
0000      86: r..bypassdos:    equ    $15 ; bypass dos check
0000      87: r..bpasintro:     equ    $ff
0000      88:
0000      89:
0000      90: l_frstscreenbyte: equ    r..bypassdos
0000      91: ;-----
0000      92: ; The language the output file is in.
0000      93: ;
0000      94: ;
0000      95: ; 0 = Undefined
0000      96: ; 1 = English
0000      97: ; 2 = German
0000      98: ;
0000      99: r..language:      equ    1
0000     100: ;-----
0000     101: ;Programming equates for the sio2usb!
0000     102: ;
0000     103: ;
0000     104: ;
0000     105: usbF_Read_Only:   equ    $01 ; Read Only
0000     106: usbF_Archive:     equ    $32 ; Archive Bit 0000x
```

```

0000 107: ;-----
0000 108: ;
0000 109: usb_Get_stat:    equ  $00  ; 4.4.1 Get U Statu
0000 110: usb_Set_RTC:     equ  $01  ; 4.4.2 Set Real Ti
0000 111: usb_Read_RTC:    equ  $02  ; 4.4.3 Read Real T
0000 112: usb_CD:          equ  $03  ; 4.4.4 Cha
0000 113: usb_RFATDIR:     equ  $04  ; 4.4.5 Read FAT Di
0000 114: usb_SET_ATARI_DRV_CFG: equ  $05  ; 4.4.6 Set ATARI d
0000 115: usb_Read_ATARI_DRV_CFG: equ  $06  ; 4.4.7 Rea
0000 116: usb_Create_ATR:  equ  $07  ; 4.4.8 Cre
0000 117: usb_Read_Free:   equ  $08  ; 4.4.9 Read amount
0000 118: usb_Delete_fd:   equ  $09  ; 4.4.10 Delete FAT
0000 119: usb_Read_VFAT_dir: equ  $0A  ; 4.4.11 Read VFAT
0000 120: usb_Rename_FAT:  equ  $0B  ; 4.4.12 Re
0000 121: ;usb_?:         equ  $0C  ; unknow
0000 122: usb_Copy_FAT_file: equ  $0D  ; 4.4.13 Copy FAT f
0000 123: usb_Change_Attrib: equ  $0F  ; 4.4.14 Change FAT
0000 124:
0000 125:
0000 126: ;-----
0000 127: ;
0000 128: pgm_sio2usb:     equ  $71    ; Program s
0000 129: siov_read:       equ  $40    ; Read
0000 130: siov_write:      equ  $80    ; write
0000 131: ;-----
0000 132: ;
0000 133: usb_disk1:       equ  1      ; Emulated
0000 134: usb_disk2:       equ  2      ; Emulated
0000 135: usb_disk3:       equ  3      ; Emulated
0000 136: usb_disk0:       equ  0      ; Emulated
0000 137: ;
0000 138: usb_timeout:     equ  7      ; sio timeo
0000 139: xfersize:        equ  128    ; Buffer si
0000 140: ;
0000 141: usb_inuse:       equ  usb_disk0 ; USB devic
0000 142: siov_do:         equ  siov_write
0000 143: usb_disk:        equ  usb_disk0 ;
0000 144: ;-----
0000 145: ZPF0:           equ  $F0
0000 146: ZPF1:           equ  ZPF0+1
0000 147: SDBASE:         equ  $700
0000 148: SDDEV:          equ  $761
0000 149: SDDATE:         equ  $77B
0000 150: SDTIME:         equ  SDDATE+3
0000 151: EDITRV:         equ  $E400
0000 152:
0000 153:
0000 154:
0000 155: .include equates
0000 156: .include globals
0000 157: .include macros
0000 158:
0000 159: .org $4000

```

```

4000      160:
4000      161: win_start:
4000      162: header_info:
4000      163:          .include      header
42a7      164: ;
42a7      165: Start: ;          jmp      parm_errors          ; d
42a7 a960  166:          lda      #$60
42a9 8da742 167:          sta      start
42ac 209655 168:          jsr      r_time
42af a50a  169:          LDA      COMTAB
42b1 38    170:          SEC
42b2 e90a  171:          SBC      # low lsio
42b4 8dd749 172:          STA      XSIO+1
42b7 a50b  173:          LDA      COMTAB+1
42b9 e900  174:          SBC      # high lsio
42bb 8dd849 175:          STA      XSIO+2
42be a96c  176:          lda      #$6c
42c0 8dd649 177:          sta      xsio
42c3      178:
42c3 a50a  179:          lda      comtab
42c5 18    180:          CLC
42c6 6903  181:          adc      # low ZCRNAME
42c8 8df543 182:          sta      cli.crunch+1
42cb a50b  183:          lda      comtab+1
42cd 6900  184:          adc      # high ZCRNAME
42cf 8df643 185:          sta      cli.crunch+2
42d2 20f443 186:          jsr      cli.crunch
42d5 d003  187:          bne      .get_parm1          ; n
42d7 4cd048 188:          jmp      go_to_it            ;go
42da      189:
42da      190: .get_parm1:
42da a50a  191:          lda      comtab
42dc 18    192:          CLC
42dd 6921  193:          adc      # low COMFNAM      ; p
42df 8582  194:          sta      s_ptr
42e1 a50b  195:          lda      comtab+1
42e3 6900  196:          adc      # high COMFNAM
42e5 8583  197:          sta      s_ptr+1
42e7      198:          mea      parm_1,d_ptr        ;ge
42e7 a9d0  ----:          lda      #low parm_1
42e9 8584  ----:          sta      d_ptr
42eb a946  ----:          lda      #high parm_1
42ed 8585  ----:          sta      d_ptr + 1
42ef 20f74f 199:          jsr      tostr              ; c
42f2      200:
42f2      201: ;          jsr      printsi
42f2      202: ;          dc.b      $9b,$9b,".1",-1
42f2      203: ;
42f2      204:          mea      parm_1,s_ptr
42f2 a9d0  ----:          lda      #low parm_1
42f4 8582  ----:          sta      s_ptr
42f6 a946  ----:          lda      #high parm_1
42f8 8583  ----:          sta      s_ptr + 1

```

```

42fa      205:          mea  set_data,d_ptr
42fa a974 ----:      lda  #low set_data
42fc 8584 ----:      sta  d_ptr
42fe a946 ----:      lda  #high set_data
4300 8585 ----:      sta  d_ptr + 1
4302 20444d 206:      jsr  substr          ; C
4305 c900  207:      cmp  #$00          ; i
4307 d003  208:      bne  .a~a          ; g
4309 4cd448 209:      jmp  menu          ; G
430c      210: .a~a:
430c 20f443 211:      jsr  cli.crunch      ; G
430f d003  212:      bne  .get_parm2     ; i
4311 4c4b49 213:      jmp  key_in        ; g
4314      214:
4314      215: .get_parm2:
4314      216: ;      jsr  printsi
4314      217: ;      dc.b  ".2",-1
4314      218:
4314 a50a  219:      lda  comtab
4316 18    220:      CLC
4317 6921  221:      adc  # low COMFNAM   ; M
4319 8582  222:      sta  s_ptr
431b a50b  223:      lda  comtab+1
431d 6900  224:      adc  # high COMFNAM  ; P
431f 8583  225:      sta  s_ptr+1
4321      226:      mea  parm_2,d_ptr    ; D
4321 a910 ----:      lda  #low parm_2
4323 8584 ----:      sta  d_ptr
4325 a947 ----:      lda  #high parm_2
4327 8585 ----:      sta  d_ptr + 1
4329 20f74f 227:      jsr  tostr          ; m
432c      228:
432c 20f443 229:      jsr  cli.crunch      ; l
432f d003  230:      bne  .get_parm3     ; y
4331      231:
4331 4c4546 232:      jmp  missing_parm3   ; M
4334      233:
4334      234: .get_parm3:
4334      235: ;      jsr  printsi
4334      236: ;      dc.b  ".3",-1
4334      237:
4334 a50a  238:      lda  comtab
4336 18    239:      CLC
4337 6921  240:      adc  # low COMFNAM   ; s
4339 8582  241:      sta  s_ptr
433b a50b  242:      lda  comtab+1
433d 6900  243:      adc  # high COMFNAM  ;
433f 8583  244:      sta  s_ptr+1
4341      245:      mea  parm_3,d_ptr    ; D
4341 a950 ----:      lda  #low parm_3
4343 8584 ----:      sta  d_ptr
4345 a947 ----:      lda  #high parm_3
4347 8585 ----:      sta  d_ptr + 1

```

```

4349 20f74f 246:      jsr  tostr          ; c
434c      247:
434c 20f443 248:      jsr  cli.crunch      ; s
434f d003  249:      bne  .exit          ; Y
4351      250:
4351 4cf84a 251:      jmp  cli_get_Time.Date ; o
4354      252: ;      jmp  .exit1
4354      253:
4354      254: .exit:
4354 a50a  255:      lda  comtab
4356 18   256:      CLC
4357 6921  257:      adc  # low COMFNAM    ; s
4359 8582  258:      sta  s_ptr
435b a50b  259:      lda  comtab+1
435d 6900  260:      adc  # high COMFNAM   ; s
435f 8583  261:      sta  s_ptr+1
4361      262:      mea  parm_4,d_ptr     ; D
4361 a990  ----:      lda  #low parm_4
4363 8584  ----:      sta  d_ptr
4365 a947  ----:      lda  #high parm_4
4367 8585  ----:      sta  d_ptr + 1
4369 20f74f 263:      jsr  tostr          ; c
436c      264:
436c      265:      mea  parm_1,s_ptr
436c a9d0  ----:      lda  #low parm_1
436e 8582  ----:      sta  s_ptr
4370 a946  ----:      lda  #high parm_1
4372 8583  ----:      sta  s_ptr + 1
4374      266:      mea  filename,d_ptr
4374 a990  ----:      lda  #low filename
4376 8584  ----:      sta  d_ptr
4378 a946  ----:      lda  #high filename
437a 8585  ----:      sta  d_ptr + 1
437c 202050 267:      jsr  toascii
437f      268:
437f      269: .exit1:
437f      270: ;      lda  consol
437f      271: ;      cmp  #consol_start    ;co
437f      272: ;      beq  .no_tell
437f      273: ;      cmp  #consol_select
437f      274: ;      beq  .give_menu
437f      275: ;      cmp  #consol_option
437f      276: ;      beq  .give_cli_info    ;an
437f      277:
437f      278:
437f 20ba4d 279:      jsr  printsi
4382 9b2043 280:      dc.b  $9b," CLI Parms"
438d 9b5061 281:      dc.b  $9b,"Parm_1 =",-1
4397      282:      mea  parm_1,pr_ptr
4397 a9d0  ----:      lda  #low parm_1
4399 8588  ----:      sta  pr_ptr
439b a946  ----:      lda  #high parm_1
439d 8589  ----:      sta  pr_ptr + 1

```

```

439f 20244d 283:      jsr  echos
43a2      284:
43a2 20ba4d 285:      jsr  printsi
43a5 9b5061 286:      dc.b  $9b,"Parm_2 =",-1
43af      287:      mea  parm_2,pr_ptr
43af a910 ----:      lda  #low parm_2
43b1 8588 ----:      sta  pr_ptr
43b3 a947 ----:      lda  #high parm_2
43b5 8589 ----:      sta  pr_ptr + 1
43b7 20244d 288:      jsr  echos
43ba      289:
43ba 20ba4d 290:      jsr  printsi
43bd 9b5061 291:      dc.b  $9b,"Parm_3 =",-1
43c7      292:      mea  parm_3,pr_ptr
43c7 a950 ----:      lda  #low parm_3
43c9 8588 ----:      sta  pr_ptr
43cb a947 ----:      lda  #high parm_3
43cd 8589 ----:      sta  pr_ptr + 1
43cf 20244d 293:      jsr  echos
43d2      294:
43d2 20ba4d 295:      jsr  printsi
43d5 9b5061 296:      dc.b  $9b,"Parm_4 =",-1
43df      297:      mea  parm_4,pr_ptr
43df a990 ----:      lda  #low parm_4
43e1 8588 ----:      sta  pr_ptr
43e3 a947 ----:      lda  #high parm_4
43e5 8589 ----:      sta  pr_ptr + 1
43e7 20244d 298:      jsr  echos
43ea 60      299:      rts
43eb      300:
43eb 4cec51 301: .no_tell:      jmp  text_to_screen      ;ge
43ee 4c4b49 302: .give_menu:      jmp  key_in          ;Ke
43f1 4cf743 303: .give_cli_info:      jmp  parm_errors
43f4      304:
43f4 4c0000 305: cli.crunch:      jmp  $0000
43f7      306:
43f7      307: parm_errors:
43f7      308:
43f7 20ba4d 309:      jsr  printsi
43fa 7d      310:      dc.b  $7d
43fb 9b      311:      dc.b  $9B
43fc 205379 312:      dc.b  " Syntax for use:",$9b
440d 205365 313:      dc.b  " SetUTime [Parm] <Date> <T
442d 203c50 314:      dc.b  " <Parm> = "
4437 20274d 315:      dc.b  " 'Menu' "
443f dbd3e5 316:      dc.b  <+128>,"[Sets Time/Date]
4452 9b      317:      dc.b  $9b
4453 202020 318:      dc.b  "      = "
445d 202748 319:      dc.b  " 'Help' "
4465 dbd4e8 320:      dc.b  <+128>,"[This Help Screen]
4478 9b      321:      dc.b  $9b
4479 202020 322:      dc.b  "      = "
4483 202749 323:      dc.b  " 'Info' "

```



```

448b dbceef 324:      dc.b  <+128>,"[Notes from Author]
449e 9b  325:      dc.b  $9b
449f 202020 326:      dc.b  "      = "
44a9 20272f 327:      dc.b  " '/'? "
44b1 dbd4e8 328:      dc.b  <+128>,"[This Screen]"
44be 9b  329:      dc.b  $9b
44bf 202020 330:      dc.b  "      = "
44c9 202752 331:      dc.b  " 'Read' "
44d1 dbd5d0 332:      dc.b  <+128>,"[UPDATE TDLine]"
44e0 9b  333:      dc.b  $9b
44e1 202020 334:      dc.b  "      = "
44eb 202753 335:      dc.b  " 'Set' "
44f3 dbd3e5 336:      dc.b  <+128>,"[Sets Time Menu]"
4503 9b  337:      dc.b  $9b
4504 202020 338:      dc.b  "      = "
450e 202753 339:      dc.b  " 'Set' "
4516 a7bcc4 340:      dc.b  <+128>,"<Date>"
451e 20  341:      dc.b  " "
451f a7bcd4 342:      dc.b  <+128>,"<Time>"
4527 20  343:      dc.b  " "
4528 9b  344:      dc.b  $9b
4529 20  345:      dc.b  " "
452a d5f0e4 346:      dc.b  <+128>,"Updates SIO2USB fro
4547 9b  347:      dc.b  $9b
4548 9b  348:      dc.b  $9b
4549 20  349:      dc.b  " "
454a c4e1f4 350:      dc.b  <+128>,"Date"
454e 20466f 351:      dc.b  " Format 'MM-DD-YY' '09-10-
456b 9b  352:      dc.b  $9b
456c 20  353:      dc.b  " "
456d d4e9ed 354:      dc.b  <+128>,"Time"
4571 20466f 355:      dc.b  " Format 'HH:MM:SS' '13:10:
458e 9b  356:      dc.b  $9b
458f 204d4d 357:      dc.b  " MM=Month 01-12 Two Number
45b0 9b  358:      dc.b  $9b
45b1 204444 359:      dc.b  " DD=Day  01-31 Two Number
45d2 9b  360:      dc.b  $9b
45d3 205959 361:      dc.b  " YY=Year  00-99 Two Number
45f4 9b9b  362:      dc.b  $9b,$9b
45f6 20  363:      dc.b  " "
45f7 c5f8e1 364:      dc.b  <+128>,"Example"
45fe 9b  365:      dc.b  $9b
45ff 205365 366:      dc.b  " SetUTime Set 09-14-12 07:
461e 9b  367:      dc.b  $9b
461f 205365 368:      dc.b  " SetUTime "
4629 ceefa0 369:      dc.b  <+128>,"No Parm"
4630 205570 370:      dc.b  " Updates TDLine2!"
4641      371:
4641 9b  372:      dc.b  $9b
4642 9b  373:      dc.b  $9b
4643 ff  374:      dc.b  -1
4644 60  375:      RTS
4645      376: missing_parm3:

```

```

4645      377:
4645 20ba4d 378:      jsr  printsi
4648 9b9b  379:      dc.b  $9B,$9B
464a 4d6973 380:      dc.b  "Missing [MM-DD-YY] and [HH
4671 9b    381:      dc.b  $9b
4672 ff    382:      dc.b  -1
4673 60    383:      rts
4674      384:
4674      385: ;~~~~~
4674      386: ;
4674      387: ; My Xio work Vars and temp registers for all ciov
4674      388: ;
4674      389: ;
4674 035345 390: set_data:      dc.b  3,"SET"
4678 044865 391: set_help:      dc.b  4,"Help"
467d 04696e 392: set_info:      dc.b  4,"info"
4682 022f3f 393: set_question:  dc.b  2,"/"
4685 045265 394: set_read:      dc.b  4,"Read"
468a 044d65 395: set_menu:      dc.b  4,"Menu"
468f      396:
468f 00    397: need_pause:    dc.b  0
4690      398: FILENAME:      ds.b  64
46d0      399:
46d0      400: parm_1:        ds.b  64
4710      401: parm_2:        ds.b  64
4750      402: parm_3:        ds.b  64
4790      403: parm_4:        ds.b  64
47d0      404: in_str:        ds.b  128
4850      405: instr:         ds.b  128
48d0      406: ;
48d0      407: ;=====
48d0      408: ;
48d0      409: ;I am going to read sio2usb time/date and set tdlne2
48d0      410: ;
48d0      411: go_to_it:
48d0 4c9950 412:      jmp   sio2usb
48d3 60    413:      rts
48d4      414:
48d4      415: ;=====
48d4      416: ;gotta check the parms and menu's help, info ect
48d4      417: ;
48d4      418: ;=====
48d4      419: ;Search the string pointed to by S_PTR for the substring
48d4      420: ;pointed to by D_PTR. Returns a 0 if substring not found.
48d4      421: ;=====
48d4      422: ;
48d4      423: menu:
48d4      424:      mea   parm_1,s_ptr
48d4 a9d0  ----:      lda   #low parm_1
48d6 8582  ----:      sta   s_ptr
48d8 a946  ----:      lda   #high parm_1
48da 8583  ----:      sta   s_ptr + 1
48dc      425:      mea   set_data,d_ptr

```

```

48dc a974 ----:    lda    #low set_data
48de 8584 ----:    sta    d_ptr
48e0 a946 ----:    lda    #high set_data
48e2 8585 ----:    sta    d_ptr + 1
48e4 20444d 426:    jsr    substr
48e7 c900 427:    cmp    #$00
48e9 f003 428:    beq    .a
48eb 4c4b49 429:    jmp    key_in      ; set time
48ee      430: .a:
48ee      431:    mea    set_help,d_ptr
48ee a978 ----:    lda    #low set_help
48f0 8584 ----:    sta    d_ptr
48f2 a946 ----:    lda    #high set_help
48f4 8585 ----:    sta    d_ptr + 1
48f6 20444d 432:    jsr    substr
48f9 c900 433:    cmp    #$00
48fb f003 434:    beq    .b
48fd 4cf743 435:    jmp    parm_errors
4900      436: .b:
4900      437:    mea    set_info,d_ptr
4900 a97d ----:    lda    #low set_info
4902 8584 ----:    sta    d_ptr
4904 a946 ----:    lda    #high set_info
4906 8585 ----:    sta    d_ptr + 1
4908 20444d 438:    jsr    substr
490b c900 439:    cmp    #$00
490d f003 440:    beq    .c
490f 4cec51 441:    jmp    text_to_screen
4912      442: .c:
4912      443:    mea    set_read,d_ptr
4912 a985 ----:    lda    #low set_read
4914 8584 ----:    sta    d_ptr
4916 a946 ----:    lda    #high set_read
4918 8585 ----:    sta    d_ptr + 1
491a 20444d 444:    jsr    substr
491d c900 445:    cmp    #$00
491f f003 446:    beq    .d
4921 4c9950 447:    jmp    sio2usb
4924      448: .d:
4924      449:    mea    set_menu,d_ptr
4924 a98a ----:    lda    #low set_menu
4926 8584 ----:    sta    d_ptr
4928 a946 ----:    lda    #high set_menu
492a 8585 ----:    sta    d_ptr + 1
492c 20444d 450:    jsr    substr
492f c900 451:    cmp    #$00
4931 f003 452:    beq    .e
4933 4c4b49 453:    jmp    key_in
4936      454: .e:
4936      455:    mea    set_question,d_ptr
4936 a982 ----:    lda    #low set_question
4938 8584 ----:    sta    d_ptr
493a a946 ----:    lda    #high set_question

```

```

493c 8585 ----:      sta  d_ptr + 1
493e 20444d 456:      jsr  substr
4941 c900  457:      cmp  #$00
4943 f003  458:      beq  .f
4945 4cf743 459:      jmp  parm_errors
4948      460: .f:
4948      461:
4948      462:
4948 4cf743 463:      jmp  parm_errors
494b      464:
494b      465: ;=====
494b      466: ;it take some work to get the sio2usb ready for
494b      467: ;read time set time
494b      468: key_in:
494b 20d949 469:      jsr  get_usb_time
494e      470:
494e a971  471:      lda  #pgm_sio2usb
4950 8d0003 472:      sta  $300          ; D
4953 a901  473:      lda  #usb_Set_RTC    ;
4955 8d0203 474:      sta  $0302          ; s
4958 a980  475:      lda  #siov_write
495a 8d0303 476:      sta  $0303
495d 206449 477:      jsr  clock_io
4960 209549 478:      jsr  clock_error
4963 60     479:      rts
4964      480:
4964      481:
4964      482:
4964 a900  483: clock_io:      lda  #usb_disk
4966 8d0103 484:      sta  $301          ; D
4969      485:
4969 a99d  486:      lda  # low usb_Time_date
496b 8d0403 487:      sta  $0304          ; D
496e a94c  488:      lda  # high usb_Time_date
4970 8d0503 489:      sta  $0305          ; D
4973      490:
4973 a907  491:      lda  #usb_timeout
4975 8d0603 492:      sta  $0306          ; T
4978 a900  493:      lda  #$00
497a 8d0703 494:      sta  $0307          ; D
497d      495:
497d      496:
497d a980  497:      lda  #low xfersize
497f 8d0803 498:      sta  $0308          ; N
4982 a900  499:      lda  #high xfersize
4984 8d0903 500:      sta  $0309          ; N
4987 a900  501:      lda  #$00
4989 8d0a03 502:      sta  $030a          ; D
498c a900  503:      lda  #$00
498e 8d0b03 504:      sta  $030b          ; D
4991 20d649 505:      jsr  xsio
4994 60     506:      rts
4995      507:

```

```

4995      508:
4995      509: clock_error:
4995 ad0303 510:          lda    $303
4998 c901   511:          cmp    #$01
499a f01a   512:          beq    ok_time_set
499c 20ba4d 513: sio2usb_error: jsr    printsi
499f 9b9b   514:          dc.b   $9b,$9b
49a1 457272 515:          dc.b   "Error $"
49a8 ff     516:          dc.b   -1
49a9 ad0303 517:          lda    $303
49ac 204750 518:          jsr    drive_error
49af 20ba4d 519:          jsr    printsi
49b2 9b9bff 520:          dc.b   $9b,$9b,-1
49b5 60     521:          rts
49b6      522:
49b6      523: ok_time_set:
49b6 20c851 524:          jsr    sio2usb.SETTIME
49b9 20ba4d 525:          jsr    printsi
49bc 9b9b   526:          dc.b   $9b,$9b
49be 53696f 527:          dc.b   "Sio2USB "
49c6 d4e9ed 528:          dc.b   <+128>,"Time Date Set"
49d3 9b     529:          dc.b   $9b
49d4 ff     530:          dc.b   -1
49d5      531: .b:
49d5 60     532:          rts
49d6      533:
49d6 4cffff 534: XSIO:      jmp    $ffff
49d9      535:
49d9      536: ;=====
49d9      537: ;read time out of sio2usb
49d9      538: ;
49d9      539: get_usb_time:
49d9 a971   540:          lda    #pgm_sio2usb
49db 8d0003 541:          sta    $300          ; D
49de a902   542:          lda    #usb_Read_RTC          ;
49e0 8d0203 543:          sta    $0302          ; s
49e3 a940   544:          lda    #siov_read
49e5 8d0303 545:          sta    $0303
49e8 206449 546:          jsr    clock_io
49eb      547:
49eb ad0303 548:          lda    $303
49ee c901   549:          cmp    #$01
49f0 f01a   550:          beq    .a
49f2 20ba4d 551:          jsr    printsi
49f5 9b9b   552:          dc.b   $9b,$9b
49f7 53494f 553:          dc.b   "SIO2USB "
49ff ceeff4 554:          dc.b   <+128>,"Not found"
4a08 ff     555:          dc.b   -1
4a09 4c9c49 556:          jmp    sio2usb_error
4a0c      557:
4a0c 20104a 558: .a:      jsr    get_Time.Date
4a0f 60     559:          rts
4a10      560: ;=====

```

```

4a10      561: ;Time Date menu
4a10      562: ;
4a10      563: ;it not in the Doc's but this is a length byte String
4a10      564: ;and you have to transfer 128 bytes not 6.
4a10      565: ;
4a10      566: ;gonna read sio2usb clock
4a10      567: ;get input from user
4a10      568: ;Set the sio2usb time date
4a10      569: ;
4a10      570:
4a10      571: get_Time.Date:
4a10      572:
4a10 20ba4d 573: .a:      Jsr  printsi
4a13 9b9b  574:      dc.b  $9b,$9b
4a15 204d6f 575:      dc.b  " Month Format 01-12",$9b
4a29 456e74 576:      dc.b  "Enter Month ==> "
4a39 ff    577:      dc.b  -1
4a3a 20b64b 578:      jsr  Ascii_dec      ; g
4a3d 8da24c 579:      sta  usb_month
4a40      580:
4a40 20ba4d 581:      Jsr  printsi
4a43 9b9b  582:      dc.b  $9b,$9b
4a45 204461 583:      dc.b  " Day Format 01-31",$9b
4a57 456e74 584:      dc.b  "Enter Day ==> "
4a67 ff    585:      dc.b  -1
4a68 20b64b 586:      jsr  Ascii_dec      ; g
4a6b 8da14c 587:      sta  usb_day
4a6e      588:
4a6e 20ba4d 589:      Jsr  printsi
4a71 9b9b  590:      dc.b  $9b,$9b
4a73 456e74 591:      dc.b  "Enter Year 20??",$9b
4a83 205965 592:      dc.b  " Year Format 00-99",$9b
4a96 456e74 593:      dc.b  "Enter Year ==> "
4aa6 ff    594:      dc.b  -1
4aa7 20b64b 595:      jsr  Ascii_dec      ; g
4aaa 8da34c 596:      sta  usb_year
4aad      597:
4aad 20ba4d 598:      Jsr  printsi
4ab0 9b9b  599:      dc.b  $9b,$9b
4ab2 456e74 600:      dc.b  "Enter Time 0 to 23",$9b
4ac5 456e74 601:      dc.b  "Enter Hour ==> "
4ad5 ff    602:      dc.b  -1
4ad6 20b64b 603:      jsr  Ascii_dec      ; g
4ad9 8da04c 604:      sta  usb_Hour
4adc      605:
4adc 20ba4d 606:      Jsr  printsi
4adf 9b    607:      dc.b  $9b
4ae0 456e74 608:      dc.b  "Enter Minute => "
4af0 ff    609:      dc.b  -1
4af1 20b64b 610:      jsr  Ascii_dec      ; g
4af4 8d9f4c 611:      sta  usb_Minutes
4af7      612:
4af7 60    613:      rts

```

```

4af8      614: ;=====
4af8      615: ;gonna read sio2usb clock
4af8      616: ;read command line interface
4af8      617: ;Set the sio2usb time date
4af8      618: ;
4af8      619: cli_get_Time.Date:
4af8 a971 620:      lda  #pgm_sio2usb
4afa 8d0003 621:      sta  $300      ; D
4afd a902 622:      lda  #usb_Read_RTC      ;
4aff 8d0203 623:      sta  $0302      ; s
4b02 a940 624:      lda  #siov_read
4b04 8d0303 625:      sta  $0303
4b07 206449 626:      jsr  clock_io
4b0a      627:
4b0a ad0303 628:      lda  $303
4b0d c901 629:      cmp  #$01
4b0f f01a 630:      beq  .a
4b11 20ba4d 631:      jsr  printsi
4b14 9b9b 632:      dc.b  $9b,$9b
4b16 53494f 633:      dc.b  "SIO2USB "
4b1e ceef4 634:      dc.b  <+128>,"Not found"
4b27 ff 635:      dc.b  -1
4b28 4c9c49 636:      jmp  sio2usb_error
4b2b      637:
4b2b      638: .a:
4b2b 20ba4d 639:      jsr  printsi
4b2e 9b2044 640:      dc.b  $9b," Date",-1
4b36 ae1447 641:      ldx  parm_2+4
4b39 ad1547 642:      lda  parm_2+5
4b3c 202c4c 643:      jsr  Ascii_dec_cli
4b3f 8da24c 644:      sta  usb_month
4b42      645:
4b42 20ba4d 646:      jsr  printsi
4b45 2dff 647:      dc.b  "-",-1
4b47 ae1747 648:      ldx  parm_2+7
4b4a ad1847 649:      lda  parm_2+8
4b4d 202c4c 650:      jsr  Ascii_dec_cli
4b50 8da14c 651:      sta  usb_day
4b53      652:
4b53 20ba4d 653:      jsr  printsi
4b56 2dff 654:      dc.b  "-",-1
4b58 ae1a47 655:      ldx  parm_2+10
4b5b ad1b47 656:      lda  parm_2+11
4b5e 202c4c 657:      jsr  Ascii_dec_cli
4b61 8da34c 658:      sta  usb_year
4b64      659:
4b64 20ba4d 660:      jsr  printsi
4b67 202054 661:      dc.b  " Time",-1
4b6f ae5147 662:      ldx  parm_3+1
4b72 ad5247 663:      lda  parm_3+2
4b75 202c4c 664:      jsr  Ascii_dec_cli
4b78 8da04c 665:      sta  usb_Hour
4b7b      666:

```

```

4b7b 20ba4d 667:      jsr  printsi
4b7e 3aff 668:      dc.b  ":", -1
4b80 ae5447 669:      ldx  parm_3+4
4b83 ad5547 670:      lda  parm_3+5
4b86 202c4c 671:      jsr  Ascii_dec_cli
4b89 8d9f4c 672:      sta  usb_Minutes
4b8c      673:
4b8c 20ba4d 674:      jsr  printsi
4b8f 3aff 675:      dc.b  ":", -1
4b91 ae5747 676:      ldx  parm_3+7
4b94 ad5847 677:      lda  parm_3+8
4b97 202c4c 678:      jsr  Ascii_dec_cli
4b9a 8d9e4c 679:      sta  usb_second
4b9d      680:
4b9d a971 681:      lda  #pgm_sio2usb
4b9f 8d0003 682:      sta  $300      ; D
4ba2 a901 683:      lda  #usb_Set_RTC      ;
4ba4 8d0203 684:      sta  $0302      ; s
4ba7 a980 685:      lda  #siov_write
4ba9 8d0303 686:      sta  $0303
4bac 206449 687:      jsr  clock_io
4baf 209549 688:      jsr  clock_error
4bb2 20c851 689:      jsr  sio2usb.SETTIME
4bb5      690: ;      jsr  sio2usb
4bb5 60 691:      rts
4bb6      692: ;~~~~~
4bb6      693: ;Ascii to Dec Conversion Routien
4bb6      694: ;input from keyboard needed!
4bb6      695: ;
4bb6      696: Ascii_dec:
4bb6 a900 697:      lda  #$00
4bb8 8d294c 698:      sta  .temp
4bbb 8d2b4c 699:      sta  .echo
4bbe      700:
4bbe 206050 701: .a:      jsr  get_key
4bc1 8d254c 702:      sta  .acc
4bc4 c92f 703:      cmp  #'0'-1
4bc6 90f6 704:      bcc  .a
4bc8 c93a 705:      cmp  #'9'+1
4bca b0f2 706:      bcs  .a
4bcc ad254c 707:      lda  .acc
4bcf 20104e 708:      jsr  echo
4bd2 ad254c 709:      lda  .acc
4bd5      710: ;      SEC
4bd5 e92f 711:      SBC  #$2f
4bd7 8d274c 712:      sta  .y
4bda a8 713:      tay
4bdb b9114c 714:      lda  .ascii_high,y
4bde 8d294c 715:      sta  .temp
4be1      716: ;      jsr  drive_error
4be1      717:
4be1 206050 718: .b:      jsr  get_key
4be4 8d264c 719:      sta  .acc1

```



```

4be7 c92f 720:      cmp    #'0'-1
4be9 90f6 721:      bcc    .b
4beb c93a 722:      cmp    #'9'+1
4bed b0f2 723:      bcs    .b
4bef ad264c 724:      lda    .acc1
4bf2 20104e 725:      jsr    echo
4bf5 ad264c 726:      lda    .acc1
4bf8      727:
4bf8      728:      ;          sec
4bf8 e92f 729:      sbc    #$2f
4bfa 8d274c 730:      sta    .y
4bfd a8 731:      tay
4bfe b91b4c 732:      lda    .ascii_low,y
4c01 4d294c 733:      eor    .temp
4c04 8d2b4c 734:      sta    .echo
4c07 20ba4d 735:      jsr    printsi
4c0a 9b9bff 736:      dc.b   $9b,$9b,-1
4c0d ad2b4c 737:      lda    .echo
4c10 60 738:      rts
4c11      739:
4c11      740:
4c11 001020 741: .Ascii_high:      dc.b   $00,$10,$20,$30,$40,$50,$60
4c1b 000102 742: .ascii_low:      dc.b   $00,$01,$02,$03,$04,$05,$06
4c25 00 743: .acc:      dc.b   0
4c26 00 744: .acc1:      dc.b   0
4c27 00 745: .y:      dc.b   0
4c28 00 746: .x:      dc.b   0
4c29 00 747: .temp:      dc.b   0
4c2a 00 748: .temp1:      dc.b   0
4c2b 00 749: .echo:      dc.b   0
4c2c      750:
4c2c      751: ;~~~~~
4c2c      752: ;Ascii to Dec Conversion Routien
4c2c      753: ;input from cli
4c2c      754: ;
4c2c      755: Ascii_dec_cli:
4c2c 8e8c4c 756:      stx    .high_byte
4c2f 8d8d4c 757:      sta    .low_byte
4c32      758:
4c32 a900 759:      lda    #$00
4c34 8d894c 760:      sta    .temp
4c37 8d8b4c 761:      sta    .echo
4c3a      762:
4c3a ad8c4c 763: .a:      lda    .high_byte
4c3d 8d854c 764:      sta    .acc
4c40 20104e 765:      jsr    echo      ;debug rout
4c43 ad854c 766:      lda    .acc
4c46 e92f 767:      SBC    #$2f
4c48 8d874c 768:      sta    .y
4c4b a8 769:      tay
4c4c b9714c 770:      lda    .ascii_high,y
4c4f 8d894c 771:      sta    .temp
4c52      772:

```

```

4c52 ad8d4c 773: .b:          lda  .low_byte
4c55 8d864c 774:          sta  .acc1
4c58 20104e 775:          jsr  echo          ;debug rout
4c5b ad864c 776:          lda  .acc1
4c5e      777:
4c5e      778: ;          sec
4c5e e92f 779:          sbc  #$2f
4c60 8d874c 780:          sta  .y
4c63 a8 781:          tay
4c64 b97b4c 782:          lda  .ascii_low,y
4c67 4d894c 783:          eor  .temp
4c6a 8d8b4c 784:          sta  .echo
4c6d      785:
4c6d      786: ;          jsr  printsi
4c6d      787: ;          dc.b  $9b,$9b,-1
4c6d ad8b4c 788:          lda  .echo
4c70 60 789:          rts
4c71      790:
4c71      791:
4c71 001020 792: .Ascii_high:      dc.b  $00,$10,$20,$30,$40,$50,$60
4c7b 000102 793: .ascii_low:      dc.b  $00,$01,$02,$03,$04,$05,$06
4c85 00 794: .acc:          dc.b  0
4c86 00 795: .acc1:         dc.b  0
4c87 00 796: .y:           dc.b  0
4c88 00 797: .x:           dc.b  0
4c89 00 798: .temp:        dc.b  0
4c8a 00 799: .temp1:       dc.b  0
4c8b 00 800: .echo:        dc.b  0
4c8c 00 801: .high_byte:   dc.b  0
4c8d 00 802: .low_byte:    dc.b  0
4c8e      803:
4c8e      804:
4c8e      805: ;-----
4c8e      806: sio2usb.READRTC:
4c8e 71 807:          dc.b  $71
4c8f 00 808:          dc.b  $00
4c90 02 809:          dc.b  $02
4c91 40 810:          dc.b  $40
4c92 9d4c 811:          dc.w  su2.buf
4c94 0700 812:          dc.w  7
4c96 8000 813:          dc.w  128
4c98 00 814:          dc.b  0
4c99 00 815:          dc.b  0
4c9a 00 816:          dc.b  0
4c9b 00 817:          dc.b  0
4c9c 00 818:          dc.b  0
4c9d      819:
4c9d      820: ;~~~~~
4c9d      821: ;Time Date Working buffer
4c9d      822: ;
4c9d      823: ;it not in the Doc's but this is a length byte String
4c9d      824: ;and you have to transfer 128 bytes not 6.
4c9d      825: ;

```

```

4c9d      826: su2.buf:
4c9d 06    827: usb_Time_date:      dc.b  6
4c9e 00    828: usb_second:        dc.b  0
4c9f 34    829: usb_Minutes:        dc.b  52
4ca0 0b    830: usb_Hour:          dc.b  11
4ca1 0d    831: usb_day:          dc.b  13
4ca2 09    832: usb_month:        dc.b  9
4ca3 0c    833: usb_year:         dc.b  12
4ca4      834: Data_buffer:      ds.b 128
4d24      835:
4d24      836:
4d24      837: ; prints: prints string with address in PR_PTR, length by
4d24      838: ; ~~~~~
4d24      839:
4d24      840: echos:
4d24 a000  841:          ldy  #0
4d26 b188  842:          lda  (pr_ptr),y
4d28 f017  843:          beq  .xpri
4d2a 8d434d 844:          sta  .maxlth
4d2d c8    845:          iny
4d2e 8c424d 846:          sty  .tempy
4d31      847:
4d31 ac424d 848: .pril:      ldy  .tempy
4d34 b188  849:          lda  (pr_ptr),y
4d36 20104e 850:          jsr  fast_output
4d39 ee424d 851: .prnext:    inc  .tempy
4d3c ce434d 852:          dec  .maxlth
4d3f d0f0  853:          bne  .pril
4d41 60    854: .xpri:      rts
4d42      855:
4d42 01    856: .tempy:      dc.b  1
4d43 01    857: .maxlth:     dc.b  1
4d44      858:
4d44      859:
4d44      860:
4d44      861: ;=====
4d44      862: ;Search the string pointed to by S_PTR for the substring
4d44      863: ;pointed to by D_PTR. Returns a 0 if substring not found.
4d44      864: ;=====
4d44      865: substr:
4d44 a000  866:          ldy  #0
4d46 b182  867:          lda  (s_ptr),y
4d48 8db84d 868:          sta  .s_lth
4d4b b184  869:          lda  (d_ptr),y
4d4d 8db94d 870:          sta  .d_lth
4d50 a582  871:          lda  s_ptr
4d52 8d744d 872:          sta  .comp+1
4d55 a583  873:          lda  s_ptr+1
4d57 8d754d 874:          sta  .comp+2
4d5a a584  875:          lda  d_ptr
4d5c 8d7d4d 876:          sta  .get+1
4d5f a585  877:          lda  d_ptr+1
4d61 8d7e4d 878:          sta  .get+2

```

```

4d64 a901 879:          lda  #1
4d66 8db74d 880:         sta  .loc
4d69 a000 881:         ldy  #0
4d6b a200 882:         ldx  #0
4d6d 209b4d 883:         jsr  .up_str
4d70 20a84d 884:         jsr  .up_sub
4d73      885:
4d73 b9ffff 886: .comp:      lda  $ffff,y
4d76 207550 887:         jsr  upcase
4d79 8db64d 888:         sta  .temp
4d7c bdfbff 889: .get:      lda  $ffff,x
4d7f 207550 890:         jsr  upcase
4d82 cdb64d 891:         cmp  .temp
4d85 f00b 892:         beq  .match
4d87 a201 893: .nomatch:  ldx  #1
4d89 eeb74d 894:         inc  .loc
4d8c 209b4d 895:         jsr  .up_str
4d8f 4c734d 896:         jmp  .comp
4d92 20a84d 897: .match:    jsr  .up_sub
4d95 209b4d 898:         jsr  .up_str
4d98 4c734d 899:         jmp  .comp
4d9b      900:
4d9b c8 901: .up_str:    iny
4d9c ccb84d 902:         cpy  .s_lth
4d9f 9006 903:         bcc  .xup
4da1 f004 904:         beq  .xup
4da3 68 905:         pla
4da4 68 906:         pla
4da5 a900 907:         lda  #0
4da7 60 908: .xup:      rts
4da8      909:
4da8 e8 910: .up_sub:    inx
4da9 ec94d 911:         cpx  .d_lth
4dac 9007 912:         bcc  .xsub
4dae f005 913:         beq  .xsub
4db0 68 914:         pla
4db1 68 915:         pla
4db2 adb74d 916:         lda  .loc
4db5 60 917: .xsub:     rts
4db6      918:
4db6 00 919: .temp:     dc.b 0
4db7 00 920: .loc:      dc.b 0
4db8 00 921: .s_lth:    dc.b 0
4db9 00 922: .d_lth:    dc.b 0
4dba      923:
4dba      924:
4dba      925: ;-----
4dba      926: ;prints Routeen macro
4dba      927: ;this is a slow print to the screen
4dba      928: ;usage jsr prints
4dba      929: ; .byte $9b," string to be printed",$ff
4dba      930: ;-----
4dba      931: ;

```

```

4dba      932: printsi:
4dba 68    933:          pla
4dbb 8dcb4d 934:          sta .pstr+1
4dbe 68    935:          pla
4dbf 8dcc4d 936:          sta .pstr+2
4dc2 eecb4d 937: .prsl:    inc .pstr+1
4dc5 d003   938:          bne .pstr
4dc7 eecc4d 939:          inc .pstr+2
4dca adffff 940: .pstr:    lda $ffff
4dcd c9ff   941:          cmp #$ff
4dcf f006   942:          beq .estri
4dd1 20104e 943:          jsr fast_output
4dd4 4cc24d 944:          jmp .prsl
4dd7 adcc4d 945: .estri:   lda .pstr+2
4dda 48    946:          pha
4ddb adcb4d 947:          lda .pstr+1
4dde 48    948:          pha
4ddf 60    949:          rts
4de0      950: ;
4de0      951: ;
4de0      952: ;
4de0      953: ;
4de0      954: ;-----
4de0      955: ;actual screen handler
4de0      956: ;screen offset definitions
4de0      957: ;-----
4de0      958: scr_offset:
4de0 0000   959:          dc.w 0
4de2 2800   960:          dc.w 40
4de4 5000   961:          dc.w 80
4de6 7800   962:          dc.w 120
4de8 a000   963:          dc.w 160
4dea c800   964:          dc.w 200
4dec f000   965:          dc.w 240
4dee 1801   966:          dc.w 280
4df0 4001   967:          dc.w 320
4df2 6801   968:          dc.w 360
4df4 9001   969:          dc.w 400
4df6 b801   970:          dc.w 440
4df8 e001   971:          dc.w 480
4dfa 0802   972:          dc.w 520
4dfc 3002   973:          dc.w 560
4dfe 5802   974:          dc.w 600
4e00 8002   975:          dc.w 640
4e02 a802   976:          dc.w 680
4e04 d002   977:          dc.w 720
4e06 f802   978:          dc.w 760
4e08 2003   979:          dc.w 800
4e0a 4803   980:          dc.w 840
4e0c 7003   981:          dc.w 880
4e0e 9803   982:          dc.w 920
4e10      983: ;
4e10      984: echo:

```

```

4e10      985: fast_output:
4e10 855a 986:      sta $5a
4e12 ad0407 987:      lda sc.redirect
4e15 c902 988:      cmp #print_p
4e17 f01f 989:      beq .go_cio
4e19      990: .leo:
4e19 adff02 991:      lda $02ff      ; CONTROL 1
4e1c c900 992:      cmp #$00
4e1e d0f9 993:      bne .leo
4e20      994:
4e20 a55a 995:      lda $5a
4e22 c920 996:      cmp #32
4e24 9004 997:      bcc .do_look
4e26 c97d 998:      cmp #125
4e28 9013 999:      bcc .do_here
4e2a a000 1000: .do_look:      ldy #0
4e2c b9334f 1001: .lookup:      lda .char_table,y
4e2f f00c 1002:      beq .do_here
4e31 c55a 1003:      cmp $5a
4e33 f003 1004:      beq .go_cio
4e35 c8 1005:      iny
4e36 d0f4 1006:      bne .lookup
4e38      1007:
4e38      1008:
4e38 a55a 1009: .go_cio:      lda $5a
4e3a 4c434f 1010:      jmp putlocal
4e3d      1011:
4e3d      1012:
4e3d a000 1013: .do_here:      ldy #0
4e3f a55d 1014:      lda $5d
4e41 915e 1015:      sta ($5e),y
4e43 a55a 1016:      lda $5a
4e45 c99b 1017:      cmp #$9b
4e47 f044 1018:      beq .docr
4e49 20124f 1019: .notcr:      jsr .get_adr
4e4c a55a 1020:      lda $5a
4e4e 297f 1021:      and #$7f
4e50 c920 1022:      cmp #32
4e52 9007 1023:      bcc .add64
4e54 c960 1024:      cmp #96
4e56 9009 1025:      bcc .sub32
4e58 4c644e 1026:      jmp .asis
4e5b 18 1027: .add64:      clc
4e5c 6940 1028:      adc #64
4e5e 4c644e 1029:      jmp .asis
4e61 38 1030: .sub32:      sec
4e62 e920 1031:      sbc #32
4e64 245a 1032: .asis:      bit $5a
4e66 1002 1033:      bpl .xxlate
4e68 0980 1034:      ora #$80
4e6a a000 1035: .xxlate:      ldy #0
4e6c 915e 1036:      sta ($5e),y
4e6e e655 1037:      inc $55

```

```

4e70 e663 1038:      inc    $63
4e72 a555 1039:      lda    $55
4e74 c928 1040:      cmp    #40
4e76 b019 1041:      bcs    .next_row
4e78 c8    1042:      iny
4e79 b15e 1043:      lda    ($5e),y
4e7b 855d 1044:      sta    $5d
4e7d 0980 1045:      ora    #$80
4e7f aef002 1046:      ldx    752
4e82 d002 1047:      bne    .nocurs1
4e84 915e 1048:      sta    ($5e),y
4e86 e65e 1049: .nocurs1:      inc    $5e
4e88 d002 1050:      bne    .nooav
4e8a e65f 1051:      inc    $5e+1
4e8c 60    1052: .nooav:      rts
4e8d a900 1053: .docr:      lda    #0
4e8f 8563 1054:      sta    $63
4e91 a552 1055: .next_row:      lda    $52
4e93 8555 1056:      sta    $55
4e95 e654 1057:      inc    $54
4e97 a454 1058:      ldy    $54
4e99 c018 1059:      cpy    #24
4e9b b013 1060:      bcs    .scroll
4e9d 20124f 1061:      jsr    .get_adr
4ea0 a000 1062:      ldy    #0
4ea2 b15e 1063:      lda    ($5e),y
4ea4 855d 1064:      sta    $5d
4ea6 aef002 1065:      ldx    752
4ea9 d004 1066:      bne    .nocurs2
4eab 0980 1067:      ora    #$80
4ead 915e 1068:      sta    ($5e),y
4eaf 60    1069: .nocurs2:      rts
4eb0      1070:
4eb0 c654 1071: .scroll:      dec    $54
4eb2 a558 1072:      lda    $58
4eb4 8568 1073:      sta    $68
4eb6 18    1074:      clc
4eb7 6928 1075:      adc    #40
4eb9 855e 1076:      sta    $5e
4ebb a559 1077:      lda    $58+1
4ebd 8569 1078:      sta    $68+1
4ebf 6900 1079:      adc    #0
4ec1 855f 1080:      sta    $5e+1
4ec3 a000 1081:      ldy    #0
4ec5 a203 1082:      ldx    #3
4ec7 b15e 1083: .sloop:      lda    ($5e),y
4ec9 9168 1084:      sta    ($68),y
4ecb c8    1085:      iny
4ecc d0f9 1086:      bne    .sloop
4ece e65f 1087:      inc    $5e+1
4ed0 e669 1088:      inc    $68+1
4ed2 ca    1089:      dex
4ed3 d0f2 1090:      bne    .sloop

```

```

4ed5 b15e 1091: .sloop2:      lda  ($5e),y
4ed7 9168 1092:                sta  ($68),y
4ed9 c8   1093:      iny
4eda c098 1094:      cpy  #152
4edc 90f7 1095:      bcc  .sloop2
4ede a558 1096:      lda  $58
4ee0 18   1097:      clc
4ee1 6998 1098:      adc  #920&$ff
4ee3 855e 1099:      sta  $5e
4ee5 a559 1100:      lda  $58+1
4ee7 6903 1101:      adc  #920/256
4ee9 855f 1102:      sta  $5e+1
4eeb a000 1103:      ldy  #0
4eed 98   1104:      tya
4eee 915e 1105: .clear:      sta  ($5e),y
4ef0 c8   1106:      iny
4ef1 c028 1107:      cpy  #40
4ef3 90f9 1108:      bcc  .clear
4ef5 aef002 1109:      ldx  752
4ef8 d006 1110:      bne  .nocurs3
4efa a980 1111:      lda  #$80
4efc a455 1112:      ldy  $55
4efe 915e 1113:      sta  ($5e),y
4f00 a900 1114: .nocurs3:      lda  #0
4f02 855d 1115:      sta  $5d
4f04 a55e 1116:      lda  $5e
4f06 18   1117:      clc
4f07 6555 1118:      adc  $55
4f09 855e 1119:      sta  $5e
4f0b a55f 1120:      lda  $5e+1
4f0d 6900 1121:      adc  #0
4f0f 855f 1122:      sta  $5e+1
4f11 60   1123:      rts
4f12 a554 1124: .get_adr:      lda  $54
4f14 0a   1125:      asl
4f15 a8   1126:      tay
4f16 a558 1127:      lda  $58
4f18 18   1128:      clc
4f19 79e04d 1129:      adc  scr_offset,y
4f1c 855e 1130:      sta  $5e
4f1e a559 1131:      lda  $58+1
4f20 79e14d 1132:      adc  scr_offset+1,y
4f23 855f 1133:      sta  $5e+1
4f25 a55e 1134:      lda  $5e
4f27 18   1135:      clc
4f28 6555 1136:      adc  $55
4f2a 855e 1137:      sta  $5e
4f2c a55f 1138:      lda  $5e+1
4f2e 6900 1139:      adc  #0
4f30 855f 1140:      sta  $5e+1
4f32 60   1141:      rts
4f33      1142: .char_table:
4f33 1b1c1d 1143:      dc.b  27,28,29,30,31,125,126,127

```



```

4f3b 9c9d9e 1144:          dc.b  156,157,158,159,253,254,255
4f43      1145:
4f43      1146:
4f43      1147: putlocal:
4f43 a20b  1148:          ldx  #11
4f45 8e4203 1149:          stx  $0342
4f48 a200  1150:          ldx  #0
4f4a 8e4803 1151:          stx  $0348
4f4d 8e4903 1152:          stx  $0349
4f50 4c56e4 1153:          jmp  $e456
4f53      1154: ;
4f53      1155:
4f53      1156:
4f53      1157:
4f53      1158:
4f53      1159: ;-----
4f53      1160: ;JSR GET_CARD
4f53      1161: ;
4f53      1162: ;-----
4f53 002020 1163: gtbyte:          dc.b  $00,"  "
4f5b 202020 1164: qstrin:          dc.b  "  "
4f62 0306  1165: gtwr:           dc.b  $03,$06
4f64      1166:
4f64      1167:
4f64      1168: get_byte:
4f64 ad624f 1169:          lda  gtwr
4f67 8d804f 1170:          sta  gt1+1
4f6a a000  1171:          ldy  #$00
4f6c 8c534f 1172:          sty  gtbyte
4f6f cd624f 1173:          cmp  gtwr
4f72 f00b  1174:          beq  gt1
4f74      1175: get_card:
4f74 ad634f 1176:          lda  gtwr+1
4f77 8d804f 1177:          sta  gt1+1
4f7a a000  1178:          ldy  #$00
4f7c 8c534f 1179:          sty  gtbyte
4f7f c0ff  1180: gt1:         cpy  #$ff
4f81 f03e  1181:          beq  gt3
4f83 206050 1182:          jsr  get_key
4f86 207550 1183:          jsr  upcase
4f89 c99b  1184:          cmp  #$9b
4f8b f034  1185:          beq  gt3
4f8d c97e  1186:          cmp  #$7e      ; BACKSPACE
4f8f f01b  1187:          beq  gt2
4f91 c930  1188:          cmp  #'0'
4f93 90ea  1189:          bcc  gt1
4f95 c93a  1190:          cmp  #':'
4f97 10e6  1191:          bpl  gt1
4f99 ac534f 1192:          ldy  gtbyte
4f9c c8     1193:          iny
4f9d 8c534f 1194:          sty  gtbyte
4fa0 99534f 1195:          sta  gtbyte,y
4fa3 20104e 1196:          jsr  echo

```

```

4fa6 c9ff 1197:      cmp    #$ff
4fa8 d0d5 1198:      bne    gt1
4faa f0d3 1199:      beq    gt1
4fac ac534f 1200: gt2:      ldy    gtbyte      ; BACKSPACE
4faf c000 1201:      cpy    #$00
4fb1 f0cc 1202:      beq    gt1
4fb3 a97e 1203:      lda    #$7e
4fb5 20104e 1204:      jsr    echo
4fb8 ce534f 1205:      dec    gtbyte
4fbb c9ff 1206:      cmp    #$ff
4fbd d0c0 1207:      bne    gt1
4fbf f0be 1208:      beq    gt1
4fc1      1209: ;
4fc1      1210: ; -MATH Stuf FOR CARD CONVERSION-
4fc1      1211: gt3:
4fc1 a99b 1212:      lda    #$9b
4fc3 ac534f 1213:      ldy    gtbyte
4fc6 c8 1214:      iny
4fc7 99534f 1215:      sta    gtbyte,y
4fca 8c534f 1216:      sty    gtbyte
4fcd a953 1217:      lda    # low gtbyte
4fcf 8582 1218:      sta    s_ptr
4fd1 a94f 1219:      lda    # high gtbyte
4fd3 8583 1220:      sta    s_ptr+1
4fd5 a95b 1221:      lda    # low qstrin
4fd7 8584 1222:      sta    d_ptr
4fd9 a94f 1223:      lda    # high qstrin
4fdb 8585 1224:      sta    d_ptr+1
4fdd 202050 1225:      jsr    toascii
4fe0 a95b 1226:      lda    # low qstrin
4fe2 85f3 1227:      sta    $f3
4fe4 a94f 1228:      lda    # high qstrin
4fe6 85f4 1229:      sta    $f3+1
4fe8 a900 1230:      lda    #$00
4fea 85f2 1231:      sta    $f2
4fec 2000d8 1232:      jsr    $d800
4fef 20d2d9 1233:      jsr    $d9d2
4ff2 a5d4 1234:      lda    $d4
4ff4 a6d5 1235:      ldx    $d4+1
4ff6 60 1236:      rts
4ff7      1237: ;
4ff7      1238: ; Make a length-byte string from an ASCII ($9b termin
4ff7      1239: ; ~~~~~
4ff7      1240:
4ff7 a000 1241: tostr:      ldy    #0
4ff9 b182 1242: .tfe:      lda    (s_ptr),y
4ffb f007 1243:      beq    .gotend
4ffd c99b 1244:      cmp    #$9b
4fff f003 1245:      beq    .gotend
5001 c8 1246:      iny
5002 d0f5 1247:      bne    .tfe
5004 8c1f50 1248: .gotend:      sty    .lth
5007 c000 1249:      cpy    #0

```

```

5009 f00c 1250:      beq  .rtn
500b a000 1251:      ldy  #0
500d b182 1252: .loop:      lda  (s_ptr),y
500f c8   1253:      iny
5010 9184 1254:      sta  (d_ptr),y
5012 cc1f50 1255:      cpy  .lth
5015 90f6 1256:      bcc  .loop
5017 a000 1257: .rtn:      ldy  #0
5019 ad1f50 1258:      lda  .lth
501c 9184 1259:      sta  (d_ptr),y
501e 60   1260:      rts
501f      1261:
501f 01   1262: .lth:      dc.b  1
5020      1263:
5020      1264: ;
5020      1265:
5020      1266: ;
5020      1267: ;  Make an ASCII ($9b terminated) string out of a lth-b
5020      1268: ; ~~~~~
5020      1269:
5020      1270: toascii:
5020 a000 1271:      ldy  #0
5022 b182 1272:      lda  (s_ptr),y      ;ge
5024 8d4650 1273:      sta  .lth
5027 f019 1274:      beq  .null
5029 c8   1275:      iny
502a ee4650 1276:      inc  .lth
502d b182 1277: .loop:      lda  (s_ptr),y
502f 88   1278:      dey
5030 9184 1279:      sta  (d_ptr),y
5032 c8   1280:      iny
5033 c8   1281:      iny
5034 cc4650 1282:      cpy  .lth
5037 d0f4 1283:      bne  .loop
5039 ac4650 1284: .done:      ldy  .lth
503c 88   1285:      dey
503d a99b 1286: .rtn:      lda  #$9b
503f 9184 1287:      sta  (d_ptr),y
5041 60   1288:      rts
5042 a000 1289: .null:      ldy  #0
5044 f0f7 1290:      beq  .rtn
5046      1291:
5046 01   1292: .lth:      dc.b  1
5047      1293:
5047      1294:
5047      1295: ;-----
5047      1296: ;  GET KEY ROUTEEN
5047      1297: ;this routeen get's one key and
5047      1298: ;return it to the accumulator
5047      1299: ;-----
5047      1300: ;
5047 48   1301: drive_error:      pha
5048 4a   1302:      lsr

```

```

5049 4a 1303:      lsr
504a 4a 1304:      lsr
504b 4a 1305:      lsr
504c 205250 1306:   jsr  .a
504f 68 1307:      pla
5050 290f 1308:      and  #$0f
5052 c90a 1309: .a:   cmp  #$0a
5054 b004 1310:      bcs  .b
5056 0930 1311:      ora  #$30
5058 d002 1312:      bne  .c
505a 6936 1313: .b:   adc  #$36
505c 20104e 1314: .c:   jsr  echo
505f 60 1315:      rts
5060      1316: ;
5060      1317: ;
5060      1318: ;-----
5060      1319: ; GET KEY ROUTEEN
5060      1320: ;this routeen get's one key and
5060      1321: ;return it to the accumulator
5060      1322: ;-----
5060      1323: ;
5060 86b6 1324: get_key:  stx  $b6
5062 84b7 1325:      sty  $b7
5064 206c50 1326:   jsr  .a
5067 a6b6 1327:      ldx  $b6
5069 a4b7 1328:      ldy  $b7
506b 60 1329:      rts
506c ad25e4 1330: .a:   lda  $e420+5
506f 48 1331:      pha
5070 ad24e4 1332:   lda  $e420+4
5073 48 1333:      pha
5074 60 1334:      rts
5075      1335: ;
5075      1336: ;-----
5075      1337: ;CHANGES THE CHARTER TO UPCASE
5075      1338: ;-----
5075      1339: upcase:
5075 c961 1340:      cmp  #$61
5077 9006 1341:      bcc  .a
5079 c97a 1342:      cmp  #$7a
507b b002 1343:      bcs  .a
507d e91f 1344:      sbc  #$1f
507f 60 1345: .a:   rts
5080      1346: ;
5080      1347: ;-----
5080      1348: ;CHANGES THE CHARTER TO LOCASE
5080      1349: ;-----
5080      1350: locase:
5080 c941 1351:      cmp  #$41
5082 9006 1352:      bcc  .a
5084 c95a 1353:      cmp  #$5a
5086 b002 1354:      bcs  .a
5088 691f 1355:      adc  #$1f

```

```

508a 60 1356: .a:          rts
508b    1357: ;-----
508b    1358: ;NO INVERSE INV_OFF
508b    1359: ;-----
508b c980 1360: inv_off:      cmp  #$80
508d 3002 1361:              bmi  .a
508f e980 1362:              sbc  #$80
5091 60 1363: .a:          rts
5092    1364: ;-----
5092    1365: ;FORCED INVERSE INV_ON
5092    1366: ;-----
5092 c980 1367: inv_on:      cmp  #$80
5094 1002 1368:              bpl  .a
5096 2980 1369:              and  #$80
5098 60 1370: .a:          rts
5099    1371: ;
5099    1372: ;
5099    1373: ;~~~~~
5099    1374: ; SIO2USB Source
5099    1375: ;~~~~~
5099    1376: sio2usb:
5099 20ba4d 1377:          jsr  printsi
509c 9b 1378:          dc.b  $9b
509d 53494f 1379:          dc.b  "SIO2USB (C) 2008 ABBUC/RAF
50b7 9b 1380:          dc.b  $9B
50b8 ff 1381:          dc.b  -1
50b9 203d51 1382: sio2usb1:  jsr  sio2usb.COUTINIT
50bc 206f51 1383:          jsr  sio2usb.GETTIME
50bf 3016 1384:          bmi  sio2usb.SIOERROR
50c1 207d51 1385:          jsr  sio2usb.TESTDATA
50c4 20c851 1386:          jsr  sio2usb.SETTIME
50c7 4cd951 1387:          jmp  sio2usb.EXIT
50ca    1388: ;-----
50ca    1389: sio2usb.ERRORMSG:
50ca 20ba4d 1390:          jsr  printsi
50cd fd 1391:          dc.b  $FD          ; B
50ce 457272 1392:          dc.b  "Error: "
50d5 ff 1393:          dc.b  $ff
50d6 60 1394:          RTS
50d7    1395: ;-----
50d7    1396: sio2usb.SIOERROR:
50d7 20ca50 1397:          jsr  sio2usb.ERRORMSG
50da 20ba4d 1398:          jsr  printsi
50dd 647572 1399:          dc.b  "during SIO Communication"
50f5 ff 1400:          dc.b  $ff
50f6 4c2451 1401:          jmp  sio2usb.EEXIT
50f9    1402: ;-----
50f9    1403: sio2usb.DATAERROR:
50f9 20ca50 1404:          jsr  sio2usb.ERRORMSG
50fc 20ba4d 1405:          jsr  printsi
50ff 77726f 1406:          dc.b  "wrong or incomplete data r
5120 ff 1407:          dc.b  $ff
5121 4c2451 1408:          jmp  sio2usb.EEXIT

```

```

5124      1409: ;-----
5124      1410: sio2usb.EEXIT:
5124 20ba4d 1411:      jsr  printsi
5127 9b    1412:      dc.b  $9B
5128 446174 1413:      dc.b  "Date/Time not set!"
513a 9b    1414:      dc.b  $9B
513b ff    1415:      dc.b  -1
513c 60    1416:      rts
513d      1417:
513d      1418:
513d      1419: sio2usb.COUTINIT:
513d a01e  1420:      ldy  #$1E
513f b91a03 1421: .1:      lda  HATABS,Y
5142 c945  1422:      cmp  #'E'
5144 f008  1423:      beq  .3
5146 88    1424:      DEY
5147 88    1425:      DEY
5148 88    1426:      DEY
5149 10f4  1427:      bpl  .1
514b 4c6551 1428:      jmp  .2
514e b91b03 1429: .3:      lda  HATABS+1,Y
5151 85f0  1430:      sta  $f0
5153 b91c03 1431:      lda  HATABS+2,Y
5156 85f1  1432:      sta  $f1
5158 a007  1433:      ldy  #7
515a b1f0  1434:      lda  ($f0),Y
515c 8d6851 1435:      sta  sio2usb.COUT2+1
515f 88    1436:      DEY
5160 b1f0  1437:      lda  ($f0),Y
5162 8d6b51 1438:      sta  sio2usb.COUT1+1
5165 60    1439: .2:      RTS
5166      1440: ;-----
5166      1441: sio2usb.COUT:
5166 aa    1442:      TAX
5167 a9f2  1443: sio2usb.COUT2:      lda  #$F2
5169 48    1444:      PHA
516a a9af  1445: sio2usb.COUT1:      lda  #$AF
516c 48    1446:      PHA
516d 8a    1447:      TXA
516e 60    1448:      RTS
516f      1449: ;-----
516f      1450: sio2usb.GETTIME:
516f a20b  1451:      ldx  #$0B
5171 bd8e4c 1452: .1:      lda  sio2usb.READRTC,X
5174 9d0003 1453:      sta  DDEVIC,X
5177 ca    1454:      DEX
5178 10f7  1455:      bpl  .1
517a 4c59e4 1456:      jmp  SIOV
517d      1457: ;-----
517d      1458: sio2usb.TESTDATA:
517d ad9d4c 1459:      lda  su2.buf
5180 c906  1460:      cmp  #6
5182 d007  1461:      bne  .3

```

```

5184 ada44c 1462:      lda  su2.buf+7
5187 c9ff 1463:      cmp  #$FF
5189 f003 1464:      beq  .2
518b 4cf950 1465: .3:      jmp  sio2usb.DATAERROR
518e      1466: ; CONVERT BCD -> DEC
518e a205 1467: .2:      ldx  #5
5190 bd9e4c 1468: .1:      lda  su2.buf+1,X
5193 29f0 1469:      AND  #$F0
5195 4a 1470:      LSR
5196 4a 1471:      LSR
5197 85f0 1472:      sta  $f0
5199 4a 1473:      LSR
519a 65f0 1474:      adc  $f0
519c fd9e4c 1475:      SBC  su2.buf+1,X
519f 49ff 1476:      EOR  #$FF
51a1 9d9e4c 1477:      sta  su2.buf+1,X
51a4 ca 1478:      DEX
51a5 10e9 1479:      bpl  .1
51a7      1480: ;
51a7 a202 1481:      ldx  #2
51a9 a00f 1482:      ldy  #$0F
51ab bda14c 1483: .4:      lda  su2.buf+4,X
51ae 910a 1484:      sta  (DOSVEC),Y
51b0 88 1485:      DEY
51b1 ca 1486:      DEX
51b2 10f7 1487:      bpl  .4
51b4      1488: ;
51b4 a010 1489:      ldy  #$10
51b6 ada04c 1490:      lda  su2.buf+3
51b9 910a 1491:      sta  (DOSVEC),Y
51bb c8 1492:      INY
51bc ad9f4c 1493:      lda  su2.buf+2
51bf 910a 1494:      sta  (DOSVEC),Y
51c1 c8 1495:      INY
51c2 ad9e4c 1496:      lda  su2.buf+1
51c5 910a 1497:      sta  (DOSVEC),Y
51c7 60 1498:      RTS
51c8      1499: ;-----
51c8      1500: sio2usb.SETTIME:
51c8 ad01d3 1501:      lda  PORTB
51cb 48 1502:      PHA
51cc 29fe 1503:      AND  #$FE
51ce 8d01d3 1504:      sta  PORTB
51d1 20c3ff 1505:      jsr  VSETTD
51d4 68 1506:      PLA
51d5 8d01d3 1507:      sta  PORTB
51d8 60 1508:      RTS
51d9      1509: ;-----
51d9      1510: ; FALLTHROUGH
51d9      1511: ;-----
51d9      1512: sio2usb.EXIT:
51d9 20ba4d 1513:      jsr  printsi
51dc 446174 1514:      dc.b  "Date/Time set"

```

```

51e9 9bff 1515:          dc.b  $9B,$ff
51eb      1516: ; FALLTHROUGH
51eb      1517: ;-----
51eb      1518: sio2usb.TODOS:
51eb 60   1519:          rts
51ec      1520:
51ec      1521: ;=====
51ec      1522: ;
51ec      1523: ;Info screen
51ec      1524: ;
51ec      1525:
51ec      1526: text_to_screen:
51ec a901 1527:          lda  #$01
51ee 8df002 1528:          sta  752
51f1      1529: ;          jsr  start-3
51f1      1530:
51f1      1531: ;=====
51f1      1532: ;          1111111112222222
51f1      1533: ;          12345678901234567890123456
51f1 20ba4d 1534:          jsr  printsi
51f4      1535:
51f4 9b   1536:          dc.b  $9b
51f5 496620 1537:          dc.b  "If you would like the sour
521b 207468 1538:          dc.b  " this file please contact
5241 ad1c40 1539:          lda  f.Authors_name
5244 8588 1540:          sta  pr_ptr
5246 ad1d40 1541:          lda  f.Authors_name+1
5249 8589 1542:          sta  pr_ptr+1
524b 20244d 1543:          jsr  echos
524e      1544:
524e 20ba4d 1545:          jsr  printsi
5251 209bff 1546:          dc.b  " ",$9b,-1
5254      1547:
5254 ad1a40 1548:          lda  f.Command_name
5257 8588 1549:          sta  pr_ptr
5259 ad1b40 1550:          lda  f.Command_name+1
525c 8589 1551:          sta  pr_ptr+1
525e 20244d 1552:          jsr  echos
5261      1553:
5261 20ba4d 1554:          jsr  printsi
5264 204361 1555:          dc.b  " Can ONLY be distributed a
5280 706172 1556:          dc.b  "part of the RealDos Constr
52a6 526561 1557:          dc.b  "RealDos can not be a part
52c7      1558:
52c7 70726f 1559:          dc.b  "program sold profit!",$9b
52dc 696e63 1560:          dc.b  "including shareware disk's
52f8 d4e8e5 1561:          dc.b  <+128>,"The Three exception
5317 9b   1562:          dc.b  $9b
5318 c1f4e1 1563:          dc.b  <+128>,"AtariMax and/or Ste
5335 9b   1564:          dc.b  $9b
5336 c1c2c2 1565:          dc.b  <+128>,"ABBUC User Group an
535b 9b   1566:          dc.b  $9b
535c d2e5e7 1567:          dc.b  <+128>,"Registered to WASEO

```



```

5380 9b9b 1568:      dc.b  $9b,$9b
5382 ff 1569:      dc.b  -1
5383 1570:
5383 206255 1571:     jsr  pause
5386 20ba4d 1572:     jsr  printsi
5389 9b2053 1573:     dc.b  $9b," Should a company wish
53ad 492077 1574:     dc.b  "I will provide a site lice
53d1 636f73 1575:     dc.b  "cost $100.00 US Dollars!",
53ea 417461 1576:     dc.b  "Atari usergroups may use i
5415 666f72 1577:     dc.b  "for there BBS Systems and
5438 746865 1578:     dc.b  "the month.", $9b,-1
5444 1579:
5444 206255 1580:     jsr  pause
5447 20ba4d 1581:     jsr  printsi
544a 9b9b 1582:     dc.b  $9b,$9b
544c 496620 1583:     dc.b  "If you wish to have the so
546b 636f64 1584:     dc.b  "code then contact:", $9B,$9
547f 202020 1585:     dc.b  " Stephen J. Carden",
5498 202020 1586:     dc.b  " 3321 Jonathan Cir."
54b2 202020 1587:     dc.b  " Augusta, Ga. 30906"
54cc 205061 1588:     dc.b  " Paypal 'sjcardensr@bellso
54f0 1589: ;
54f0 4f7220 1590:     dc.b  "Or send a donation of $10.
551b 66696c 1591:     dc.b  "file the above address.", $
5534 1592:
5534 20ba4d 1593:     jsr  printsi
5537 9b9b 1594:     dc.b  $9b,$9b
5539 205072 1595:     dc.b  " Press "
5540 dbd2e5 1596:     dc.b  <+128>,"[Return]"
5548 20616e 1597:     dc.b  " any other key ", $9B,-1
5559 206050 1598:     jsr  get_key
555c a900 1599:     lda  #$00
555e 8df002 1600:     sta  752
5561 60 1601:     rts
5562 1602: ;
5562 1603: pause: ; a more command
5562 1604: more:
5562 20ba4d 1605:     jsr  printsi
5565 dba0ad 1606:     dc.b  <+128>,"[ - MORE - ]"
5571 ff 1607:     dc.b  -1
5572 206050 1608:     jsr  get_key
5575 c97e 1609:     cmp  #$7e ; back spac
5577 f015 1610:     beq  .esc_more
5579 c91b 1611:     cmp  #$1b ;es
557b f011 1612:     beq  .esc_more
557d 1613: ; cmp  #$1c ;up
557d 1614: ; beq  .esc_more
557d 1615:
557d 20ba4d 1616:     jsr  printsi
5580 7e7e7e 1617:     dc.b  $7e,$7e,$7e,$7e
5584 7e7e7e 1618:     dc.b  $7e,$7e,$7e,$7e
5588 7e7e7e 1619:     dc.b  $7e,$7e,$7e,$7e
558c ff 1620:     dc.b  -1

```

```

558d 60 1621:      rts
558e      1622:
558e      1623: .esc_more:
558e a900 1624:      lda  #$00
5590 8df002 1625:      sta  752
5593 68 1626:      pla
5594 68 1627:      pla
5595 60 1628:      rts
5596      1629:
5596      1630: ;
5596      1631: ;
5596      1632: r_time:
5596 a514 1633:      lda  $14
5598 c514 1634: .r_1:      cmp  $14
559a f0fc 1635:      beq  .r_1
559c ad01d3 1636:      lda  portb
559f 48 1637:      pha
55a0 29fe 1638:      and  #$fe
55a2 8d01d3 1639:      sta  portb
55a5 18 1640:      clc
55a6 a001 1641:      ldy  #$01
55a8 20c6ff 1642:      jsr  vtdon
55ab b005 1643:      bcs  r_time_not
55ad 68 1644:      pla
55ae 8d01d3 1645:      sta  portb
55b1 60 1646:      rts
55b2      1647: ;
55b2      1648: r_time_not:
55b2 68 1649:      pla
55b3 8d01d3 1650:      sta  portb
55b6 20a442 1651:      jsr  start-3
55b9 20ba4d 1652:      jsr  printsi
55bc 9b9b 1653:      dc.b  $9b,$9b
55be 209192 1654:      dc.b  " ,,,,,,,,,,,,,,,,,,,,,,"
55df 20fc 1655:      dc.b  " ü"
55e1 a0d2ad 1656:      dc.b  <+128>," R-time Handler not
55fe fc9b 1657:      dc.b  "ü",$9B
5600 20fc 1658:      dc.b  " ü"
5602 a0c9ee 1659:      dc.b  <+128>," Install tdline2.co
561f fc9b 1660:      dc.b  "ü",$9B
5621 20fc 1661:      dc.b  " ü"
5623 a0f4f2 1662:      dc.b  <+128>," trying to read the
5640 fc9b 1663:      dc.b  "ü",$9B
5642 20fc 1664:      dc.b  " ü"
5644 a0e4e1 1665:      dc.b  <+128>," date from the SIO2
5661 fc9b 1666:      dc.b  "ü",$9B
5663 20fc 1667:      dc.b  " ü"
5665 a0e1ee 1668:      dc.b  <+128>," and updating your
5682 fc9b 1669:      dc.b  "ü",$9B
5684 20fc 1670:      dc.b  " ü"
5686 a0a0a0 1671:      dc.b  <+128>," Press Reset to
56a3 fc9b 1672:      dc.b  "ü",$9B
56a5 209a92 1673:      dc.b  " §,,,,,,,,,,,,,,,,,,,,,"

```

```
56c8 68 1674:          pla
56c9 68 1675:          pla
56ca 60 1676:          rts
56cb    1677:
56cb    1678: ;
56cb    1679: ;
56cb    1680:
56cb    1681: win_end:      ds.b  0
56cb    1682: ;
56cb    1683: ;
```

End assembly: no errors