


```
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: ;      sio2ide.com
0000 9: ;
0000 10: ;This is a program is for testing a detection routeen
0000 11: ;
0000 12: ;
0000 13: ;*****
0000 14: ; Copyright (C), 2001 MMSOft, All rights reserved
0000 15: ;*****
0000 16: ;Parts of this file are copyright MMSOft & ILS
0000 17: ;*****
0000 18: ;
0000 19: ; SOURCE FILE: SIOSRV.C
0000 20: ;
0000 21: ; MODULE NAME: SIOSRV
0000 22: ;
0000 23: ; PURPOSE:   SIO service.
0000 24: ;
0000 25: ; AUTHOR:    Marek Mikolajewski (MM)
0000 26: ;
0000 27: ; convert from c to assemble BY:Stephen J. Carden
0000 28: ;
0000 29: ; HISTORY:   Ver  Date    Sign  Description
0000 30: ;
0000 31: ;      001  24-01-2002 MM    Created
0000 32: ;
0000 33: ;      init  9/10/2012   sjc new design
0000 34: ;Converted c code to assembler and Added the sio2ide info
0000 35: ;that Marek Mikolajewski sent me
0000 36: ;
0000 37: ;-----
0000 38: ;
0000 39: com_file_name:      .macro
0000 40: ;      dc.b  13,"filename.ext",$9b
0000 41: ;      dc.b  12,"SIO2IDE.com ", $9b
0000 42: ;      .endm
0000 43: ;
0000 44: ;
0000 45: ;-----
0000 46: ; File revision history. Version Number in hex
0000 47: ;
0000 48: file_ver:          equ  $10
0000 49: ;-----
0000 50: ; the Month in dec for the first time this revision
0000 51: ;was compiled
0000 52: ;
0000 53: c_month:           equ  9
```

```

0000 54: ;-----
0000 55: ; the day in dec for the first time this revision
0000 56: ;was compiled
0000 57: c_day:          equ 12
0000 58: ;-----
0000 59: ; the year in dec for the first time this revision
0000 60: ;was compiled
0000 61: c_year:          equ 2012
0000 62: ;-----
0000 63: ; Type of compiler used in program
0000 64: ;
0000 65: ; 0 = unknown Compiler
0000 66: ; 1 = xasm
0000 67: ; 2 = mac_65
0000 68: ; 3 = basic
0000 69: ; 4 = compiled basic xl
0000 70: ; 5 = C65
0000 71: ; 6 = Action
0000 72: ;
0000 73: ; We can add more as time goes on!
0000 74: ;
0000 75: ;
0000 76: ;
0000 77: xasm:            equ 1
0000 78: mac_65:          equ 2
0000 79: basic:           equ 3
0000 80: basicxl:         equ 4
0000 81: c65:            equ 5
0000 82: action:          equ 6
0000 83: ;
0000 84: file_compiler:    equ xasm
0000 85: ;-----
0000 86: ; is this relocatable code ?
0000 87: ; anyother value other than 1 or 2 would be unknown
0000 88: r..yes:           equ 1
0000 89: r..no:            equ 2
0000 90: ;
0000 91: ;-----
0000 92: ; Gotta define if it can be relocated
0000 93: l_relocatable:    equ r..no
0000 94: ;
0000 95: ;
0000 96: ;-----
0000 97: ; Gotta define if it can be relocated
0000 98: r..crl:           equ $7d
0000 99: r..crlf:          equ $9b
0000 100: r..space:         equ $20
0000 101: r..bypassdos:     equ $15 ; bypass dos check
0000 102: r..bpasintro:     equ $ff
0000 103:
0000 104:
0000 105: l_frstscreenbyte: equ r..crl
0000 106: ;-----

```

```

0000 107: ; The language the output file is in.
0000 108: ;
0000 109: ;
0000 110: ; 0 = Undefined
0000 111: ; 1 = English
0000 112: ; 2 = German
0000 113: ;
0000 114: r.language: equ 1
0000 115: ;-----
0000 116: ; SIOV commands that SIO2IDE use
0000 117: ;
0000 118: ; SIO commands accepted by SIO2IDE FS
0000 119: ;
0000 120: pgm_sio2ide: equ $71 ; Program sio2ide i
0000 121: SIOC_NONE: equ $00 ;
0000 122: SIOC_GCFG: equ $4E ; Fixed Config
0000 123: SIOC_PCFG: equ $4F ; Dummy Only ACK
0000 124: SIOC_STAT: equ $53 ; Fixed Status
0000 125: SIOC_PUT: equ $50 ; Put data
0000 126: SIOC_PUTV: equ $57 ; Put data with ver
0000 127: SIOC_GET: equ $52 ; Get data
0000 128: SIOC_FORMAT: equ $21 ; Format (always OK)
0000 129: SIOC_HS: equ $3F ; ? US Doubler
0000 130: ;
0000 131: SIOC_GETDSK: equ $11 ; Get the Disk desc
0000 132: SIOC_PUTDSK: equ $12 ; Put the Disk desc
0000 133: SIOC_FATR: equ $13 ; Get First ATR fil
0000 134: SIOC_NATR: equ $14 ; Get Next ATR file
0000 135: SIOC_OFFDSK: equ $15 ; Switch Off select
0000 136: SIOC_SCFG: equ $16 ; Save Config file
0000 137: SIOC_GETDRV: equ $17 ; Get Drive paramet
0000 138: SIOC_GETFSI: equ $18 ; Get FS info
0000 139: SIOC_FDIR: equ $19 ; Get First Dir ent
0000 140: SIOC_NDIR: equ $1A ; Get Next Dir entr
0000 141: SIOC_SCDIR: equ $1B ; Set Current worki
0000 142: SIOC_GCDIR: equ $1C ; Get Current worki
0000 143: SIOC_ICDIR: equ $1D ; Initialise Curren
0000 144: ;
0000 145: SIOC_SSECT: equ $61 ; Set Physical Disk
0000 146: ;
0000 147: SIOC_GSECT: equ $62 ; Get Physical Disk
0000 148: SIOC_PSECT: equ $63 ; Put Physical Disk
0000 149: SIOC_GLFN: equ $64 ; Get Long File Nam
0000 150: SIOC_CDRV: equ $65 ; Change Current Dr
0000 151: ;
0000 152: ide_master: equ $01 ; Master IDE Device
0000 153: ide_slave: equ $02 ; Slave IDE Device
0000 154: ;-----
0000 155: ;
0000 156: siov_read: equ $40 ; Read
0000 157: siov_write: equ $80 ; write
0000 158: ;-----
0000 159: ;

```

```

0000    160: ide_disk1: equ 1      ; Emulated Disk 1
0000    161: ide_disk2: equ 2      ; Emulated Disk 2
0000    162: ide_disk3: equ 3      ; Emulated Disk 3
0000    163: ide_disk4: equ 4      ; Emulated Disk 4
0000    164: ide_disk5: equ 5      ; Emulated Disk 5
0000    165: ide_disk6: equ 6      ; Emulated Disk 6
0000    166: ide_disk7: equ 7      ; Emulated Disk 7
0000    167: ide_disk8: equ 8      ; Emulated Disk 8
0000    168: ;
0000    169: ide_timeout: equ 20    ; sio timeout
0000    170: xfersize: equ 15      ; Buffer size we ar
0000    171: ;
0000    172: ide_inuse: equ ide_master ; Ide we are progra
0000    173: siov_do: equ siov_read
0000    174: ide_disk: equ ide_disk1 ; Aux1
0000    175: ;-----
0000    176:
0000    177:
0000    178:
0000    179:
0000    180: .include equates
0000    181: .include globals
0000    182: .include macros
0000    183:
4000    184: .org $4000
4000    185:
4000    186: win_start:
4000    187: header_info:
4000    188: .include header ; t
42a7    189:
42a7    190: ;
42a7    191: Start:
42a7    192:
42a7    193: get_sector_count:
42a7 a50a 194: LDA COMTAB ; G
42a9 38 195: SEC
42aa e90a 196: SBC # low lsio ; S
42ac 8d5643 197: STA XSIO+1 ; S
42af a50b 198: LDA COMTAB+1 ; n
42b1 e900 199: SBC # high lsio ; S
42b3 8d5743 200: STA XSIO+2 ; s
42b6 a96c 201: lda #$6c ; H
42b8 8d5543 202: sta xsio ; W
42bb 203:
42bb 204:
42bb 205: sio2ide:
42bb a901 206: lda #ide_master
42bd 8d0103 207: sta $301 ; D
42c0 201d43 208: jsr get_sio2ide
42c3 209:
42c3 205543 210: jsr xsio
42c6 ad0303 211: lda $303
42c9 c901 212: cmp #$01

```

```

42cb f002 213:      beq  .a
42cd d01d 214:      bne  .b
42cf      215:
42cf 205847 216: .a:      jsr  printsi
42d2 53696f 217:      dc.b  "Sio2ide "
42da cde1f3 218:      dc.b  <+128>,"Master"
42e0 204465 219:      dc.b  " Detected!",$9b
42eb ff 220:      dc.b  -1
42ec      221: .b:
42ec      222:
42ec a902 223:      lda  #ide_slave
42ee 8d0103 224:      sta  $301          ; D
42f1 201d43 225:      jsr  get_sio2ide
42f4      226:
42f4 205543 227:      jsr  xsio
42f7 ad0303 228:      lda  $303
42fa c901 229:      cmp  #$01
42fc f002 230:      beq  .aa
42fe d01c 231:      bne  .bb
4300      232:
4300 205847 233: .aa:      jsr  printsi
4303 53696f 234:      dc.b  "Sio2ide "
430b d3ece1 235:      dc.b  <+128>,"Slave"
4310 204465 236:      dc.b  " Detected!",$9b
431b ff 237:      dc.b  -1
431c 60 238: .bb:      rts
431d      239:
431d      240:
431d a971 241: get_sio2ide:      lda  #pgm_sio2ide
431f 8d0003 242:      sta  $300          ; D
4322      243:
4322 a911 244:      lda  #SIOC_GETDSK          ; s
4324 8d0203 245:      sta  $0302          ; D
4327 a940 246:      lda  #siov_do          ; $
4329 8d0303 247:      sta  $0303          ; D
432c      248:
432c a958 249:      lda  # low Data_buffer
432e 8d0403 250:      sta  $0304          ; D
4331 a943 251:      lda  # high Data_buffer
4333 8d0503 252:      sta  $0305          ; D
4336      253:
4336 a914 254:      lda  #ide_timeout
4338 8d0603 255:      sta  $0306          ; T
433b a900 256:      lda  #$00
433d 8d0703 257:      sta  $0307          ; D
4340      258:
4340      259:
4340 a90f 260:      lda  #Low xfersize
4342 8d0803 261:      sta  $0308          ; N
4345 a900 262:      lda  #high xfersize
4347 8d0903 263:      sta  $0309          ; N
434a      264:
434a a901 265:      lda  #ide_disk

```

```

434c 8d0a03 266:          sta  $030a          ; D
434f a900 267:          lda  #$00
4351 8d0b03 268:          sta  $030b          ; D
4354 60 269:          rts
4355 270:
4355 271:
4355 272:
4355 273:
4355 274:
4355 4cffff 275: XSIO:          jmp  $ffff
4358 276:
4358 277: ;~~~~~
4358 278: ;Working buffer
4358 279: ;
4358 280: Data_buffer:
4358 281:          ds.b 1024
4758 282:
4758 283:
4758 284:
4758 285: ;-----
4758 286: ;prints Routine macro
4758 287: ;this is a slow print to the screen
4758 288: ;usage jsr prints
4758 289: ; .byte $9b," string to be printed",$ff
4758 290: ;-----
4758 291: ;
4758 292: prints:
4758 68 293:          pla
4759 8d6947 294:          sta  .pstr+1
475c 68 295:          pla
475d 8d6a47 296:          sta  .pstr+2
4760 ee6947 297: .prsl:    inc  .pstr+1
4763 d003 298:          bne  .pstr
4765 ee6a47 299:          inc  .pstr+2
4768 adffff 300: .pstr:    lda  $ffff
476b c9ff 301:          cmp  #$ff
476d f006 302:          beq  .estri
476f 20ae47 303:          jsr  fast_output
4772 4c6047 304:          jmp  .prsl
4775 ad6a47 305: .estri:   lda  .pstr+2
4778 48 306:          pha
4779 ad6947 307:          lda  .pstr+1
477c 48 308:          pha
477d 60 309:          rts
477e 310: ;
477e 311: ;
477e 312:
477e 313: ;
477e 314: ;-----
477e 315: ;actual screen handler
477e 316: ;screen offset definitions
477e 317: ;-----
477e 318: scr_offset:

```

```

477e 0000 319:          dc.w  0
4780 2800 320:          dc.w  40
4782 5000 321:          dc.w  80
4784 7800 322:          dc.w 120
4786 a000 323:          dc.w 160
4788 c800 324:          dc.w 200
478a f000 325:          dc.w 240
478c 1801 326:          dc.w 280
478e 4001 327:          dc.w 320
4790 6801 328:          dc.w 360
4792 9001 329:          dc.w 400
4794 b801 330:          dc.w 440
4796 e001 331:          dc.w 480
4798 0802 332:          dc.w 520
479a 3002 333:          dc.w 560
479c 5802 334:          dc.w 600
479e 8002 335:          dc.w 640
47a0 a802 336:          dc.w 680
47a2 d002 337:          dc.w 720
47a4 f802 338:          dc.w 760
47a6 2003 339:          dc.w 800
47a8 4803 340:          dc.w 840
47aa 7003 341:          dc.w 880
47ac 9803 342:          dc.w 920
47ae      343: ;
47ae      344: fast_output:
47ae 855a 345:          sta  $5a
47b0 ad0407 346:          lda  sc.redirect
47b3 c902 347:          cmp  #print_p
47b5 f01f 348:          beq  .go_cio
47b7      349: .leo:
47b7 adff02 350:          lda  $02ff      ; CONTROL 1
47ba c900 351:          cmp  #$00
47bc d0f9 352:          bne  .leo
47be      353:
47be a55a 354:          lda  $5a
47c0 c920 355:          cmp  #32
47c2 9004 356:          bcc  .do_look
47c4 c97d 357:          cmp  #125
47c6 9013 358:          bcc  .do_here
47c8 a000 359: .do_look:          ldy  #0
47ca b9d148 360: .lookup:          lda  .char_table,y
47cd f00c 361:          beq  .do_here
47cf c55a 362:          cmp  $5a
47d1 f003 363:          beq  .go_cio
47d3 c8 364:          iny
47d4 d0f4 365:          bne  .lookup
47d6      366:
47d6      367:
47d6 a55a 368: .go_cio:          lda  $5a
47d8 4ce148 369:          jmp  putlocal
47db      370:
47db      371:

```



```

47db a000 372: .do_here:      ldy  #0
47dd a55d 373:                lda  $5d
47df 915e 374:                sta  ($5e),y
47e1 a55a 375:                lda  $5a
47e3 c99b 376:                cmp  #$9b
47e5 f044 377:                beq  .docr
47e7 20b048 378: .notcr:          jsr  .get_adr
47ea a55a 379:                lda  $5a
47ec 297f 380:                and  #$7f
47ee c920 381:                cmp  #32
47f0 9007 382:                bcc  .add64
47f2 c960 383:                cmp  #96
47f4 9009 384:                bcc  .sub32
47f6 4c0248 385:                jmp  .asis
47f9 18 386: .add64:          clc
47fa 6940 387:                adc  #64
47fc 4c0248 388:                jmp  .asis
47ff 38 389: .sub32:          sec
4800 e920 390:                sbc  #32
4802 245a 391: .asis:          bit  $5a
4804 1002 392:                bpl  .xxlate
4806 0980 393:                ora  #$80
4808 a000 394: .xxlate:        ldy  #0
480a 915e 395:                sta  ($5e),y
480c e655 396:                inc  $55
480e e663 397:                inc  $63
4810 a555 398:                lda  $55
4812 c928 399:                cmp  #40
4814 b019 400:                bcs  .next_row
4816 c8 401:          iny
4817 b15e 402:                lda  ($5e),y
4819 855d 403:                sta  $5d
481b 0980 404:                ora  #$80
481d aef002 405:                ldx  752
4820 d002 406:                bne  .nocurs1
4822 915e 407:                sta  ($5e),y
4824 e65e 408: .nocurs1:        inc  $5e
4826 d002 409:                bne  .nooav
4828 e65f 410:                inc  $5e+1
482a 60 411: .nooav:          rts
482b a900 412: .docr:          lda  #0
482d 8563 413:                sta  $63
482f a552 414: .next_row:       lda  $52
4831 8555 415:                sta  $55
4833 e654 416:                inc  $54
4835 a454 417:                ldy  $54
4837 c018 418:                cpy  #24
4839 b013 419:                bcs  .scroll
483b 20b048 420:                jsr  .get_adr
483e a000 421:                ldy  #0
4840 b15e 422:                lda  ($5e),y
4842 855d 423:                sta  $5d
4844 aef002 424:                ldx  752

```

```

4847 d004 425:      bne  .nocurs2
4849 0980 426:      ora  #$80
484b 915e 427:      sta  ($5e),y
484d 60   428: .nocurs2:      rts
484e      429:
484e c654 430: .scroll:      dec  $54
4850 a558 431:      lda  $58
4852 8568 432:      sta  $68
4854 18   433:      clc
4855 6928 434:      adc  #40
4857 855e 435:      sta  $5e
4859 a559 436:      lda  $58+1
485b 8569 437:      sta  $68+1
485d 6900 438:      adc  #0
485f 855f 439:      sta  $5e+1
4861 a000 440:      ldy  #0
4863 a203 441:      ldx  #3
4865 b15e 442: .sloop:      lda  ($5e),y
4867 9168 443:      sta  ($68),y
4869 c8   444:      iny
486a d0f9 445:      bne  .sloop
486c e65f 446:      inc  $5e+1
486e e669 447:      inc  $68+1
4870 ca   448:      dex
4871 d0f2 449:      bne  .sloop
4873 b15e 450: .sloop2:      lda  ($5e),y
4875 9168 451:      sta  ($68),y
4877 c8   452:      iny
4878 c098 453:      cpy  #152
487a 90f7 454:      bcc  .sloop2
487c a558 455:      lda  $58
487e 18   456:      clc
487f 6998 457:      adc  #920&$ff
4881 855e 458:      sta  $5e
4883 a559 459:      lda  $58+1
4885 6903 460:      adc  #920/256
4887 855f 461:      sta  $5e+1
4889 a000 462:      ldy  #0
488b 98   463:      tya
488c 915e 464: .clear:      sta  ($5e),y
488e c8   465:      iny
488f c028 466:      cpy  #40
4891 90f9 467:      bcc  .clear
4893 aef002 468:      ldx  752
4896 d006 469:      bne  .nocurs3
4898 a980 470:      lda  #$80
489a a455 471:      ldy  $55
489c 915e 472:      sta  ($5e),y
489e a900 473: .nocurs3:      lda  #0
48a0 855d 474:      sta  $5d
48a2 a55e 475:      lda  $5e
48a4 18   476:      clc
48a5 6555 477:      adc  $55

```

```

48a7 855e 478:      sta  $5e
48a9 a55f 479:      lda  $5e+1
48ab 6900 480:      adc  #0
48ad 855f 481:      sta  $5e+1
48af 60 482:      rts
48b0 a554 483: .get_adr:      lda  $54
48b2 0a 484:      asl
48b3 a8 485:      tay
48b4 a558 486:      lda  $58
48b6 18 487:      clc
48b7 797e47 488:      adc  scr_offset,y
48ba 855e 489:      sta  $5e
48bc a559 490:      lda  $58+1
48be 797f47 491:      adc  scr_offset+1,y
48c1 855f 492:      sta  $5e+1
48c3 a55e 493:      lda  $5e
48c5 18 494:      clc
48c6 6555 495:      adc  $55
48c8 855e 496:      sta  $5e
48ca a55f 497:      lda  $5e+1
48cc 6900 498:      adc  #0
48ce 855f 499:      sta  $5e+1
48d0 60 500:      rts
48d1 501: .char_table:
48d1 1b1c1d 502:      dc.b 27,28,29,30,31,125,126,127
48d9 9c9d9e 503:      dc.b 156,157,158,159,253,254,255
48e1 504:
48e1 505:
48e1 506: putlocal:
48e1 a20b 507:      ldx  #11
48e3 8e4203 508:      stx  $0342
48e6 a200 509:      ldx  #0
48e8 8e4803 510:      stx  $0348
48eb 8e4903 511:      stx  $0349
48ee 4c56e4 512:      jmp  $e456
48f1 513: ;
48f1 514:
48f1 515:
48f1 516:
48f1 517:
48f1 518:
48f1 519:
48f1 520:
48f1 521:
48f1 522: win_end:      ds.b 0
48f1 523: ;
48f1 524: ;

```

End assembly: no errors