


```

0000 1: ;-----
0000 2: ;      Copyright 2010 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: ;      gdevice.com
0000 9: ;
0000 10: ; This is a program sets up a graphic driver for dot
0000 11: ;      init 4/6/2010      sjc new design
0000 12: ;
0000 13: ;
0000 14: ;Updated the screen Print Utility so it will work with the
0000 15: ; This change will also help "Print P:"      8/1
0000 16: ;
0000 17: ;
0000 18: ;Also added code for Control+1 pause of screen to the high
0000 19: ;handling.      8/24/2012      sjc
0000 20: ;
0000 21: ;
0000 22: ;
0000 23: ;Added code so this file cannot run on the AtariWin800 Emu
0000 24: ;      8/27/2012      sjc
0000 25: ;
0000 26: ;
0000 27: ;
0000 28: ;
0000 29: ; ~~~~~
0000 30: ;-----
0000 31: ; Notes: o This source code MAY NOT be placed for download
0000 32: ;      o "mea" is a macro that loads the address of the
0000 33: ;      into the pointer specified by the second field.
0000 34: ;-----
0000 35: ; Assembler: MADMAC (tm) ST Cross Assembler (Atari Corp)
0000 36: ;      XASM st and IBM VERSIONS
0000 37: ;-----
0000 38: ;
0000 39: com_file_name:      .macro
0000 40: ;      dc.b 13,"filename.ext",$9b
0000 41: ;      dc.b 12,"Gdevice.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 $11
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 8

```

```
0000 54: ;~~~~~
0000 55: ; the day in dec for the first time this revision
0000 56: ;was compiled
0000 57: c_day:          equ 27
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..yes
0000 94: ;
0000 95: ;
0000 96: ;~~~~~
0000 97: ; Gotta define our Header control Byte
0000 98: ;$ff=bypass intro
0000 99: ;$15 = Bypass Dos check
0000 100: r..crl:          equ $7d
0000 101: r..crlf:         equ $9b
0000 102: r..space:        equ $20
0000 103: r..bypassDOS:    equ $15
0000 104: r..bypass:       equ $ff
0000 105:
0000 106: l_frstscreenbyte: equ r..space
```

```

0000 107: ;~~~~~
0000 108: ; The language the output file is in.
0000 109: ;
0000 110: ;
0000 111: ; 0 = Undefined
0000 112: ; 1 = English
0000 113: ; 2 = German
0000 114: ;
0000 115: r..language: equ 1
0000 116: ;~~~~~
0000 117:
0000 118:
0000 119:
0000 120: .include equates
0000 121: .include globals
0000 122: .include macros
0000 123:
0000 124:
0000 125:
4000 126: .ORG $4000
4000 127:
4000 128: win_start:
4000 129: header_info:
4000 130: .include header ; t
42a7 131: ; b
42a7 ad1fd0 132: lda consol ; c
42aa c907 133: cmp #7 ; l
42ac f003 134: beq .a ; n
42ae 4c574f 135: jmp help_menu ; S
42b1 136: .a:
42b1 ad0bd4 137: lda vcount
42b4 d0fb 138: bne .a
42b6 139:
42b6 140: ;~~~~~
42b6 141: ;
42b6 142: ; Check to see if we have a R: Handler
42b6 143: ;
42b6 144: check_r_handler:
42b6 206b56 145: jsr os_off
42b9 146: mea tsr,s_ptr
42b9 a9f5 ----: lda #low tsr
42bb 8582 ----: sta s_ptr
42bd a9ff ----: lda #high tsr
42bf 8583 ----: sta s_ptr + 1
42c1 a91a 147: lda #gdevice
42c3 208156 148: jsr seclvl ; c
42c6 304d 149: bmi .aa
42c8 20d254 150: jsr printsi
42cb 9b 151: dc.b $9b
42cc 20 152: dc.b " "
42cd bdbdbe 153: dc.b <+128>,"==>"
42d0 c7baa0 154: dc.b <+128>,"G: Handler already
42ea bcbdbd 155: dc.b <+128>,"<==>"

```

```

42ed 9b 156:          dc.b  $9b
42ee 20 157:          dc.b  " "
42ef bdbdbe 158:        dc.b  <+128>,"==">
42f2 c1eee4 159:        dc.b  <+128>,"And Registered with
430c bcbdbd 160:        dc.b  <+128>,"<==">
430f 9bff 161:        dc.b  $9b,-1
4311 207656 162:        jsr  os_on
4314 60 163:          rts          ; w
4315 164:
4315 165:
4315 166: .aa:
4315 207656 167:        jsr  os_on
4318 168:
4318 169:
4318 170: ;-----
4318 171: ;   Check to see if we are running on an emulator
4318 172: ;
4318 173: pc_run:
4318 a000 174:          ldy  #$00
431a b91a03 175: loop_1:      lda  $031a,y
431d c948 176:          cmp  #'H' ; WAS THE EDITOR
431f f00d 177:          beq  get_vec
4321 c8 178:          iny
4322 c8 179:          iny
4323 c8 180:          iny
4324 c024 181:          cpy  #36
4326 d0f2 182:          bne  loop_1
4328 a000 183:          ldy  #$00
432a 4c8043 184:          jmp  ok_look
432d 60 185:          rts
432e 186: ;
432e 187: get_vec
432e 20d254 188:          jsr  printsi
4331 9b20 189:          dc.b  $9b,$20
4333 c5f2f2 190:          dc.b  <+128>,"Error"
4338 9b 191:          dc.b  $9b
4339 204764 192:          dc.b  " Gdevice.com does not work
4359 204174 193:          dc.b  " Atari800Win PC Emulator D
437c ff 194:          dc.b  -1
437d a001 195:          ldy  #$01
437f 196:
437f 60 197:          RTS
4380 198: ;
4380 199: ;
4380 200:
4380 201: ok_look:
4380 202:
4380 203:
4380 204:
4380 a000 205:          ldy  #$00
4382 b91a03 206: laap_1:      lda  $031a,y
4385 c947 207:          cmp  #'G'
4387 f00b 208:          beq  get1_vec

```

```

4389 c8 209:      iny
438a c8 210:      iny
438b c8 211:      iny
438c c024 212:    cpy #36
438e d0f2 213:    bne laap_1
4390 4cb943 214:   jmp do_install
4393 60 215:      rts
4394      216: ;
4394      217: get1_vec:
4394 20d254 218:      JSR PRINTSI
4397 20473a 219:      dc.b " G: handler Already Instal
43b5 9b9b 220:      dc.b $9b,$9b
43b7 ff 221:      dc.b -1
43b8 60 222:      RTS
43b9      223: ;
43b9      224: do_install:
43b9      225: ;
43b9      226: ;
43b9      227: ;
43b9 ade702 228:      lda memlo ; s
43bc 8d4d4f 229:      sta segtab+dorgadr ; P
43bf 8d534f 230:      sta segtab+blkdes ; P
43c2 ade802 231:      lda memlo+1 ; s
43c5 8d4e4f 232:      sta segtab+dorgadr+1 ; P
43c8 8d544f 233:      sta segtab+blkdes+1 ; P
43cb      234:
43cb 20404d 235:      jsr relocate ; o
43ce      236:
43ce      237:
43ce a50a 238:      lda DOSVEC
43d0 8dcf47 239: w01:      sta L298B
43d3 a50b 240:      lda DOSVEC+1
43d5 8dd047 241: w02:      sta L298B+1
43d8      242:
43d8 a50c 243:      lda DOSINI
43da 8d1544 244: w03:      sta L25D0+1
43dd 8dd147 245: w04:      sta L298D
43e0 a50d 246:      lda DOSINI+1
43e2 8d1644 247: w05:      sta L25D0+2
43e5 8dd247 248: w06:      sta L298d+1
43e8 a914 249: lb01:      lda # low l25D0
43ea 850c 250:      sta DOSINI
43ec a944 251: hb01:      lda # high l25d0
43ee 850d 252:      sta DOSINI+1
43f0 201744 253: w07:      jsr reinit
43f3      254:
43f3 20d254 255:      jsr printsi ; P
43f6 9b9b 256:      dc.b $9b,$9b
43f8 20473a 257:      dc.b " G: Handler installed!",$9
4410      258:
4410 202457 259:      jsr set_g_handler
4413      260:
4413 60 261:      rts

```

```

4414      262: ;-----
4414      263: ;
4414      264: ; This gets relocated to memlow
4414      265: ;
4414      266: ;
4414      267: initz:
4414      268: ;
4414 20ffff 269: l25d0:      JSR   $FFFF
4417      270: reinit:
4417 a200   271:      ldx   #0
4419 bd1a03 272: l25d5:      LDA   $031A,X
441c f005   273:      beq   L25DF
441e e8     274:      INX
441f e8     275:      INX
4420 e8     276:      INX
4421 d0f6   277:      bne   L25D5
4423 a947   278: l25df:      LDA   #'G'
4425 9d1a03 279:      sta   $031A,X
4428 a945   280: lb02:      lda   # low g_handler
442a 9d1b03 281:      sta   $031A+1,X
442d a944   282: hb02:      lda   # high g_handler
442f 9d1c03 283:      sta   $031A+2,X
4432 a954   284: lb03:      lda   # low l2610
4434 850a   285:      sta   DOSVEC
4436 a944   286: hb03:      lda   # high l2610
4438 850b   287:      sta   DOSVEC+1
443a a940   288: lb04:      lda   # low l2eFC
443c 8de702 289:      sta   $02E7
443f a94d   290: hb04:      lda   # high l2Efc
4441 8de802 291:      sta   $02E7+1
4444 60     292:      RTS
4445      293: ;
4445      294: g_handler:
4445 6044   295: w09:      dc.w  p.open-1      ;op
4447 cb44   296: w10:      dc.w  p.close-1     ;cl
4449 e244   297: w11:      dc.w  p.getbyte-1   ;Ge
444b e344   298: w12:      dc.w  p.putbyte-1   ;Pu
444d e044   299: w13:      dc.w  p.status-1    ;St
444f 1146   300: w14:      dc.w  p.xio-1       ;XI
4451      301:
4451      302: l260d:
4451 4ce144 303: w15:      jmp   p.status
4454      304: ;
4454 a203   305: l2610:      LDX   #3
4456      306: L2612:
4456 bdcf47 307: w16:      LDA   L298B,X
4459 950a   308: w17:      sta   DOSVEC,X
445b ca     309:      DEX
445c 10f8   310:      BPL   L2612
445e 6c0a00 311:      jmp   ($000A)
4461      312:
4461      313: ;
4461 08     314: p.open:      PHP

```

```

4462 8a 315:      TXA
4463 4a 316:      lsr
4464 4a 317:      lsr
4465 4a 318:      lsr
4466 4a 319:      lsr
4467 a8 320:      TAY
4468 88 321:      DEY
4469 3004 322:     bmi L262B
446b c007 323:     cpy #7
446d 9004 324:     bcc L262F
446f a086 325: L262B: LDY #134
4471 d057 326:     bne L2686
4473 a621 327: L262F: LDX $21
4475 ca 328:      DEX
4476 8eed47 329: w18: stx L29A9
4479 bd2348 330: w19: lda L29DF,X
447c 991c48 331: w20: sta L29D8,Y
447f a52b 332:      lda $2b
4481 c9ff 333:      cmp #$FF
4483 d002 334:      bne L2643
4485 a900 335:      lda #0
4487 c903 336: L2643: CMP #3
4489 b004 337:      bcs L264B
448b aa 338:      TAX
448c bd2748 339: w21: lda L29E3,X
448f 340: L264B:
448f 8d3b48 341: w22: STA L29F7
4492 a522 342:      lda $22
4494 c940 343:      cmp #$40
4496 d017 344:      bne L266B
4498 a52a 345:      lda $2A
449a c902 346:      cmp #2
449c b004 347:      bcs L265E
449e a91c 348:      lda #$1C
44a0 d00a 349:      bne L2668
44a2 c903 350: L265E: CMP #3
44a4 b004 351:      bcs L2666
44a6 a90e 352:      lda #$0E
44a8 d002 353:      bne L2668
44aa a901 354: L2666: LDA #1
44ac 355: L2668:
44ac 8d3848 356: w23: STA L29F4
44af 357: L266B:
44af 207a47 358: w24: JSR L2936
44b2 8def47 359: w25: sta L29AB
44b5 a20a 360:      ldx #$0A
44b7 361: L2673:
44b7 bd3148 362: w26: LDA L29ED,X
44ba 9dc003 363: w27: sta $03C0,X
44bd ca 364:      DEX
44be 10f7 365:      bpl L2673
44c0 208547 366: w28: jsr L2941
44c3 3005 367:      bmi L2686

```



```

44c5 205447 368: w29:      jsr  L2910
44c8 a001 369:      ldy  #1
44ca 28 370: L2686:      PLP
44cb 60 371:      RTS
44cc      372: ;
44cc      373: p.close:
44cc 207a47 374: w30:      jsr  L2936
44cf a91b 375:      lda  #$1B
44d1 8dc003 376:      sta  $03C0
44d4 a940 377:      lda  #$40
44d6 8dc103 378:      sta  $03C0+1
44d9 a99b 379:      lda  #$9B
44db 8dc203 380:      sta  $03C0+2
44de 208547 381: w31:      jsr  L2941
44e1      382: p.status:
44e1 a001 383:      ldy  #1
44e3      384: p.getbyte:
44e3 60 385:      RTS
44e4      386: ;
44e4      387: p.putbyte
44e4 08 388:      PHP
44e5 8ee947 389: w32:      stx  L29A5
44e8 48 390:      PHA
44e9 a207 391:      ldx  #7
44eb b5d4 392: L26A7:      LDA  $d4,X
44ed 9ddf47 393: w33:      sta  L299B,X
44f0 ca 394:      DEX
44f1 10f8 395:      bpl  L26A7
44f3 ade747 396: w34:      lda  L29A3
44f6 85d4 397:      sta  $d4
44f8 ade847 398: w35:      lda  L29A3+1
44fb 85d5 399:      sta  $d4+1
44fd 68 400:      PLA
44fe acef47 401: w36:      ldy  L29AB
4501 f013 402:      beq  L26D2
4503 c040 403:      cpy  #64 ;$40
4505 d016 404:      bne  L26D9
4507 a52b 405:      lda  $2b
4509 8dec47 406: w37:      sta  L29A8
450c a9fc 407: lb05:      lda  # low l29B8
450e 85d6 408:      sta  $d6
4510 a947 409: hb05:      lda  # high l29b8
4512 85d7 410:      sta  $d6+1
4514 d038 411:      bne  L270A
4516 c99b 412: L26D2:      CMP  #$9B
4518 d003 413:      bne  L26D9
451a 4ca145 414: w38:      jmp  L275D
451d      415: ;
451d a000 416: L26D9:      LDY  #0
451f 8cec47 417: w39:      sty  L29A8
4522 84d7 418:      sty  $d6+1
4524 0a 419:      asl
4525 6eec47 420: w40:      ROR  L29A8

```

```

4528 4a 421:      lsr
4529 acef47 422: w41:      ldy L29AB
452c d00f 423:      bne L26F9
452e c920 424:      cmp #$20
4530 b004 425:      bcs L26F2
4532 6940 426:      adc #$40
4534 9007 427:      bcc L26F9
4536 c960 428: L26F2:      CMP #$60
4538 b003 429:      bcs L26F9
453a 38 430:      SEC
453b e920 431:      sbc #$20
453d a003 432: L26F9:      LDY #3
453f 0a 433: L26FB:      ASL
4540 26d7 434:      ROL $d6+1
4542 88 435:      DEY
4543 d0fa 436:      bne L26FB
4545 85d6 437:      sta $d6
4547 adf402 438:      lda $02F4
454a 05d7 439:      ORA $d6+1
454c 85d7 440:      sta $d6+1
454e a007 441: L270A:      LDY #7
4550 8cea47 442: w42:      sty L29A6
4553 b1d6 443: L270F:      LDA ($d6),Y
4555 acec47 444: w43:      ldy L29A8
4558 f002 445:      beq L2718
455a 49ff 446:      EOR #$FF
455c a007 447: L2718:      LDY #7
455e 4a 448: L271A:      lsr
455f 48 449:      PHA
4560 b1d4 450:      lda ($d4),Y
4562 6a 451:      ROR
4563 91d4 452:      sta ($d4),Y
4565 68 453:      PLA
4566 88 454:      DEY
4567 10f5 455:      bpl L271A
4569 cee47 456: w44:      dec L29A6
456c acea47 457: w45:      ldy L29A6
456f 10e2 458:      bpl L270F
4571 a007 459:      ldy #7
4573 b1d4 460: L272F:      LDA ($d4),Y
4575 c99b 461:      cmp #$9B
4577 d004 462:      bne L2739
4579 a997 463:      lda #$97
457b 91d4 464:      sta ($d4),Y
457d 88 465: L2739:      DEY
457e 10f3 466:      bpl L272F
4580 18 467:      CLC
4581 a5d4 468:      lda $d4
4583 6908 469:      adc #8
4585 8de747 470: w46:      sta L29A3
4588 85d4 471:      sta $d4
458a 9005 472:      bcc L274D
458c eee847 473: w47:      inc L29A3+1

```

```

458f e6d4 474:          inc    $d4
4591      475: L274D:
4591 ee3e48 476: w48:          INC    L29FA
4594 aeed47 477: w49:          ldx    L29A9
4597 f06b 478:          beq    L27C0
4599 ad3e48 479: w50:          lda    L29FA
459c dd2a48 480: w51:          cmp    L29E6,X
459f d063 481:          bne    L27C0
45a1 a000 482: L275D:          LDY    #0
45a3 a99b 483:          lda    #$9B
45a5 91d4 484:          sta    ($d4),Y
45a7 8cee47 485: w52:          sty    L29AA
45aa a203 486:          ldx    #3
45ac      487: L2768:
45ac 0e3e48 488: w53:          ASL    L29FA
45af 2e3f48 489: w54:          rol    L29FB
45b2 ca 490:          DEX
45b3 d0f7 491:          bne    L2768
45b5 adef47 492: w55:          lda    L29AB
45b8 c940 493:          cmp    #$40
45ba d004 494:          bne    L277C
45bc a94c 495:          lda    #$4C
45be d00c 496:          bne    L2788
45c0      497: L277C:
45c0 ade947 498: w56:          LDA    L29A5
45c3 4a 499:          lsr
45c4 4a 500:          lsr
45c5 4a 501:          lsr
45c6 4a 502:          lsr
45c7 aa 503:          TAX
45c8 ca 504:          DEX
45c9 bd1c48 505: w57:          lda    L29D8,X
45cc      506: L2788:
45cc 8d3d48 507: w58:          STA    L29F9
45cf a93c 508: lb06:          lda    # low l29f8
45d1 85da 509:          sta    $da
45d3 a948 510: hb06:          lda    # high l29f8
45d5 85db 511:          sta    $da+1
45d7      512: L2793:
45d7 207a47 513: w59:          JSR    L2936
45da a8 514:          TAY
45db b1da 515: L2797:          LDA    ($da),Y
45dd 99c003 516:          sta    $03C0,Y
45e0 c99b 517:          cmp    #$9B
45e2 d005 518:          bne    L27A5
45e4 8dee47 519: w60:          sta    L29AA
45e7 f010 520:          beq    L27B5
45e9 c8 521: L27A5:          INY
45ea c028 522:          cpy    #$28
45ec d0ed 523:          bne    L2797
45ee 18 524:          CLC
45ef a5da 525:          lda    $da
45f1 6928 526:          adc    #40      ;$28

```

```

45f3 85da 527:      sta  $da
45f5 9002 528:      bcc  L27B5
45f7 e6db 529:      inc  $da+1
45f9      530: L27B5:
45f9 208547 531: w61:      JSR  L2941
45fc adee47 532: w62:      lda  L29AA
45ff f0d6 533:      beq  L2793
4601 205447 534: w63:      jsr  L2910
4604 a207 535: L27C0:      LDX  #7
4606      536: L27C2:
4606 bddf47 537: w64:      LDA  L299B,X
4609 95d4 538:      sta  $d4,X
460b ca 539:      DEX
460c 10f8 540:      bpl  L27C2
460e 28 541:      PLP
460f a001 542:      ldy  #1
4611 60 543:      RTS
4612      544: ;
4612      545: p.xio:
4612 8ee947 546: w65:      stx  L29A5
4615 206144 547: w66:      jsr  p.open
4618 c001 548:      cpy  #1
461a d051 549:      bne  L2829
461c a904 550:      lda  #4
461e 8ded47 551: w67:      sta  L29A9
4621 a558 552:      lda  SAVMSC
4623 85d8 553:      sta  $d8
4625 8df647 554: w68:      sta  L29B2
4628 8df447 555: w69:      sta  L29B0
462b a559 556:      lda  SAVMSC+1
462d 85d9 557:      sta  $d8+1
462f 8df747 558: w70:      sta  L29B2+1
4632 8df547 559: w71:      sta  L29B0+1
4635 a522 560:      lda  $22
4637 8def47 561: w72:      sta  L29AB
463a c940 562:      cmp  #64      ;xio #64 $4
463c f032 563:      beq  L282C
463e c911 564:      cmp  #17      ;xio #17 $1
4640 b004 565:      bcs  L2802
4642 a918 566:      lda  #$18
4644 d002 567:      bne  L2804
4646 e910 568: L2802:      SBC  #$10
4648      569: L2804:
4648 8df147 570: w73:      STA  L29AD
464b a900 571: L2807:      LDA  #0
464d 8df047 572: w74:      sta  L29AC
4650      573: L280C:
4650 acf047 574: w75:      LDY  L29AC
4653 b1d8 575:      lda  ($D8),Y
4655 aee947 576: w76:      ldx  L29A5
4658 20e444 577: w77:      jsr  p.putbyte
465b eef047 578: w78:      inc  L29AC
465e adf047 579: w79:      lda  L29AC

```

```
4661 c928 580:      cmp  #40      ;$28
4663 d0eb 581:      bne  L280C
4665 209347 582: w80:      jsr  L294F
4668 cef147 583: w81:      dec  L29AD
466b d0de 584:      bne  L2807
466d      585: L2829:
466d 4ccc44 586: w82:      JMP  p.close
4670      587: ;
4670 a52a 588: L282C:      LDA  $2A
4672 c902 589:      cmp  #2
4674 900d 590:      bcc  L283F
4676 c903 591:      cmp  #3
4678 b004 592:      bcs  L283A
467a a905 593:      lda  #5
467c d002 594:      bne  L283C
467e a906 595: L283A:      LDA  #6
4680      596: L283C:
4680 8ded47 597: w83:      STA  L29A9
4683 a521 598: L283F:      LDA  $21
4685 c903 599:      cmp  #3
4687 900b 600:      bcc  L2850
4689 8dfb47 601: w84:      sta  L29B7
468c a2a0 602:      ldx  #$A0
468e a000 603:      ldy  #0
4690 a930 604:      lda  #$30
4692 d00b 605:      bne  L285B
4694 a900 606: L2850:      LDA  #0
4696 8dfb47 607: w85:      sta  L29B7
4699 a240 608:      ldx  #$40
469b a001 609:      ldy  #1
469d a918 610:      lda  #$18
469f      611: L285B:
469f 8ef847 612: w86:      STX  L29B4
46a2 8cf947 613: w87:      sty  L29B4+1
46a5 8dfa47 614: w88:      sta  L29B6
46a8 a900 615:      lda  #0
46aa 8df247 616: w89:      sta  L29AE
46ad a900 617: L2869:      LDA  #0
46af 8df347 618: w90:      sta  L29AF
46b2      619: L286E:
46b2 adf647 620: w91:      LDA  L29B2
46b5 85d8 621:      sta  $d8
46b7 adf747 622: w92:      lda  L29B2+1
46ba 85d9 623:      sta  $d8+1
46bc a200 624:      ldx  #0
46be a000 625: L287A:      LDY  #0
46c0 b1d8 626:      lda  ($d8),Y
46c2 9dfc47 627: w93:      sta  L29B8,X
46c5 acfb47 628: w94:      ldy  L29B7
46c8 f003 629:      beq  L2889
46ca 9dfd47 630: w95:      sta  L29B9,X
46cd      631: L2889:
46cd 209347 632: w96:      JSR  L294F
```

```

46d0 e8 633:      INX
46d1 c003 634:      cpy  #3
46d3 9001 635:      bcc  L2892
46d5 e8 636:      INX
46d6 e008 637: L2892:      CPX  #8
46d8 d0e4 638:      bne  L287A
46da a52a 639:      lda  $2A
46dc c902 640:      cmp  #2
46de 9030 641:      bcc  L28CC
46e0 209f47 642: w97:      jsr  L295B
46e3 a207 643:      ldx  #7
46e5      644: L28A1:
46e5 bd0448 645: w98:      LDA  L29C0,X
46e8 9dfc47 646: w99:      sta  L29B8,X
46eb ca 647:      DEX
46ec 10f7 648:      bpl  L28A1
46ee 20e444 649: w100:     jsr  p.putbyte
46f1 a207 650:      ldx  #7
46f3      651: L28AF:
46f3 bd0c48 652: w101:     LDA  L29C8,X
46f6 9dfc47 653: w102:     sta  L29B8,X
46f9 ca 654:      DEX
46fa 10f7 655:      bpl  L28AF
46fc 20e444 656: w103:     jsr  p.putbyte
46ff a52a 657:      lda  $2A
4701 c903 658:      cmp  #3
4703 d00e 659:      bne  L28CF
4705 a207 660:      ldx  #7
4707      661: L28C3:
4707 bd1448 662: w104:     LDA  L29D0,X
470a 9dfc47 663: w105:     sta  L29B8,X
470d ca 664:      DEX
470e 10f7 665:      bpl  L28C3
4710      666: L28CC:
4710 20e444 667: w106:     JSR  p.putbyte
4713      668: L28CF:
4713 eef647 669: w107:     INC  L29B2
4716 d003 670:      bne  L28D7
4718 eef747 671: w108:     inc  L29B2+1
471b      672: L28D7:
471b eef347 673: w109:     INC  L29AF
471e adf347 674: w110:     lda  L29AF
4721 c928 675:      cmp  #$28
4723 d08d 676:      bne  L286E
4725 18 677:      CLC
4726 adf447 678: w111:     lda  L29B0
4729 6df847 679: w112:     adc  L29B4
472c 8df447 680: w113:     sta  L29B0
472f 8df647 681: w114:     sta  L29B2
4732 adf547 682: w115:     lda  L29B0+1
4735 6df947 683: w116:     adc  L29B4+1
4738 8df547 684: w117:     sta  L29B0+1
473b 8df747 685: w118:     sta  L29B2+1

```

```

473e eef247 686: w119:      inc  L29AE
4741 adf247 687: w120:      lda  L29AE
4744 cdfa47 688: w121:      cmp  L29B6
4747 f003 689:      beq  L2908
4749 4cad46 690: w122:      jmp  L2869
474c 691: ;
474c a905 692: L2908:      LDA  #5
474e 8d3848 693: w123:      sta  L29F4
4751 4ccc44 694: w124:      jmp  p.close
4754 695: ;
4754 a900 696: L2910:      LDA  #0
4756 aa 697:      TAX
4757 698: L2913:
4757 9d4048 699: w125:      STA  L29FC,X
475a 9d4049 700: w126:      sta  L2AFC,X
475d 9d404a 701: w127:      sta  L2BFC,X
4760 9d404b 702: w128:      sta  L2CFC,X
4763 9d404c 703: w129:      sta  L2DFC,X
4766 e8 704:      INX
4767 d0ee 705:      bne  L2913
4769 8d3e48 706: w130:      sta  L29FA
476c 8d3f48 707: w131:      sta  L29FB
476f a940 708: lb07:      lda  # low l29FC
4771 8de747 709: w132:      sta  L29A3
4774 a948 710: hb07:      lda  # high l29fc
4776 8de847 711: w133:      sta  L29A3+1
4779 60 712:      RTS
477a 713: ;
477a a227 714: L2936:      LDX  #40-1
477c a900 715:      lda  #0
477e 9dc003 716: L293A:      STA  $03C0,X
4781 ca 717:      DEX
4782 10fa 718:      bpl  L293A
4784 60 719:      RTS
4785 720: ;
4785 a20b 721: L2941:      LDX  #11      ;$0B
4787 722: L2943:
4787 bdd347 723: w134:      LDA  L298F,X
478a 9d0003 724:      sta  $0300,X
478d ca 725:      DEX
478e 10f7 726:      bpl  L2943
4790 4c59e4 727:      jmp  SIOV
4793 728: ;
4793 18 729: L294F:      CLC
4794 a5d8 730:      lda  $d8
4796 6928 731:      adc  #40      ;$28
4798 85d8 732:      sta  $d8
479a 9002 733:      bcc  L295A
479c e6d9 734:      inc  $d8+1
479e 60 735: L295A:      RTS
479f 736: ;
479f a217 737: L295B:      LDX  #23      ;$17
47a1 a900 738:      lda  #0

```

```

47a3      739: L295F:
47a3 9d0448 740: w135:      STA  L29C0,X
47a6 ca    741:      DEX
47a7 10fa   742:      bpl  L295F
47a9 a207   743:      ldx  #7
47ab      744:
47ab a907   745: L2967:      LDA  #7
47ad 8deb47 746: w136:      sta  L29A7
47b0 bdfc47 747: w137:      lda  L29B8,X
47b3 4a     748: L296F:      lsr
47b4 48     749:      PHA
47b5 a42a   750:      ldy  $2A
47b7 08     751: L2973:      PHP
47b8 7e0448 752: w138:      ror  l29c0,x
47bb 7e0c48 753: w139:      ror  l29c8,x
47be 7e1448 754: w140:      ror  l29d0,x
47c1 28     755:      PLP
47c2 88     756:      DEY
47c3 d0f2   757:      bne  L2973
47c5 68     758:      PLA
47c6 ceeb47 759: w141:      dec  L29A7
47c9 10e8   760:      bpl  L296F
47cb ca     761:      DEX
47cc 10dd   762:      bpl  L2967
47ce 60     763:      RTS
47cf      764: ;
47cf      765: ;
47cf 0000   766: L298B:      dc.w  0
47d1 0000   767: L298d:      dc.w  0
47d3      768: ;
47d3      769: ;
47d3 400157 770: L298F:      dc.b  $40,$01,$57,$80,$c0,$03,$05
47df      771: ;
47df 000000 772: l299b:      dc.b  $00,$00,$00,$00,$00,$00,$00
47e7      773:
47e7 0000   774: L29A3:      dc.w  0
47e9 00     775: L29A5:      dc.b  0
47ea 00     776: L29A6:      dc.b  0
47eb      777:
47eb 00     778: L29A7      dc.b  0
47ec 00     779: L29A8:      dc.b  $00
47ed 00     780: L29A9:      dc.b  $00
47ee 00     781: L29AA:      dc.b  $00
47ef 00     782: L29AB:      dc.b  $00
47f0 00     783: L29AC:      dc.b  $00
47f1 00     784: L29AD:      dc.b  $00
47f2 00     785: L29AE:      dc.b  $00
47f3 00     786: L29AF:      dc.b  $00
47f4 0000   787: L29B0:      dc.w  0
47f6 0000   788: L29B2:      dc.w  0
47f8 0000   789: L29B4:      dc.w  00
47fa      790:
47fa 00     791: L29B6:      dc.b  $00

```



```

47fb 00 792: L29B7:      dc.b  $00
47fc 00 793: L29B8:      dc.b  $00
47fd 000000 794: L29B9:      dc.b  $00,$00,$00,$00
4801 000000 795:      dc.b  $00,$00,$00
4804 796:
4804 797: L29C0:      ds.b  8
480c 798:
480c 799: L29C8:      ds.b  8
4814 800:
4814 801: L29D0:      ds.b  8
481c 802:
481c 803: L29D8:      ds.b  7
4823 804:
4823 4c4c4b 805: L29DF:      dc.b  $4c,$4c,$4b,$4b
4827 806: ;
4827 08090c 807: L29E3:      dc.b  $08,$09,$0c
482a 808:
482a 002626 809: L29E6:      dc.b  $00,$26,$26,$3c,$28,$50,$78
4831 810:
4831 1b401b 811: L29ED:      dc.b  $1B,$40,$1b
4834 4e 812:      dc.b  $4e
4835 813:
4835 051b 814:      dc.b  $05,$1b
4837 4d 815:      dc.b  $4d
4838 816:
4838 051b 817: L29F4:      dc.b  $05,$1b
483a 41 818:      dc.b  $41
483b 819:
483b 08 820: l29f7:      dc.b  $08
483c 821:
483c 1b 822: l29f8:      dc.b  $1b
483d 823:
483d 00 824: L29F9:      dc.b  $00
483e 00 825: L29FA:      dc.b  $00
483f 00 826: L29FB:      dc.b  $00
4840 827: ;
4840 828: L29FC:      ds.b  256
4940 829: l2afc:      ds.b  256
4a40 830: L2BFC:      ds.b  256
4b40 831: L2CFC:      ds.b  256
4c40 832: L2DFC:      ds.b  256
4d40 833: l2efc:      ds.b  0
4d40 834: ;
4d40 835: ;
4d40 836: ;
4d40 837: ;   End of handler
4d40 838: ;
4d40 839: zend:      ds.b  0
4d40 840: ;
4d40 841: ;
4d40 842: ;
4d40 843: ;~~~~~
4d40 844: ;

```

```

4d40      845: ;          *** END OF HANDLER ***
4d40      846: ;~~~~~
4d40      847: ;
4d40      848: ;
4d40      849: ; Relocator: main entry
4d40      850: ; -----
4d40      851: ; in:
4d40      852: ; segtab = table of segment descriptors
4d40      853: ; +00 = relocater table address
4d40      854: ; +02 = originate address of block
4d40      855: ; +04 = destination originate of block
4d40      856: ; +06 = address of block
4d40      857: ; +08 = number of bytes in block
4d40      858: ; +10 = destination address of block
4d40      859: ;
4d40      860: ; reldtab = relocater table
4d40      861: ; list of address of words to adjust 2
4d40      862: ; list of address low bytes to adjust 2
4d40      863: ; list of address high bytes to adjust 3
4d40      864: ; followed by their low bytes
4d40      865: ;~~~~~
4d40 a900 866: relocate: lda #0 ; set ptr t
4d42 8d044e 867: sta segment
4d45      868:
4d45 ae044e 869: segloop: ldx segment
4d48 bd494f 870: lda segtab+rettad,x ; get addre
4d4b 8dea4d 871: sta relget+1
4d4e bd4a4f 872: lda segtab+rettad+1,x
4d51 8deb4d 873: sta relget+2
4d54 0dea4d 874: ora relget+1 ; if zero,
4d57 d001 875: bne havseg
4d59 60 876: rts
4d5a      877:
4d5a 38 878: havseg: sec ; c
4d5b bd4d4f 879: lda segtab+dorgadr,x
4d5e fd4b4f 880: sbc segtab+orgadr,x
4d61 8d054e 881: sta zoffset
4d64 bd4e4f 882: lda segtab+dorgadr+1,x
4d67 fd4c4f 883: sbc segtab+orgadr+1,x
4d6a 8d064e 884: sta zoffset+1
4d6d      885:
4d6d 20f54d 886: zwordlp: jsr getzwp
4d70 f013 887: beq zlbytlp ; if zero,
4d72 b1d7 888: lda (zwptr),y ; add offse
4d74 18 889: clc
4d75 6d054e 890: adc zoffset
4d78 91d7 891: sta (zwptr),y
4d7a c8 892: iny
4d7b b1d7 893: lda (zwptr),y
4d7d 6d064e 894: adc zoffset+1
4d80 91d7 895: sta (zwptr),y
4d82 4c6d4d 896: jmp zwordlp
4d85      897:

```

```

4d85 20f54d 898: zlbytlp:  jsr  getzwp          ; get addre
4d88 f00b 899:          beq  zhbytlp          ; if zero,
4d8a b1d7 900:          lda  (zwptr),y
4d8c 18 901:          clc
4d8d 6d054e 902:          adc  zoffset
4d90 91d7 903:          sta  (zwptr),y
4d92 4c854d 904:          jmp  zlbytlp
4d95 905:
4d95 20f54d 906: zhbytlp:  jsr  getzwp          ; get addre
4d98 f011 907:          beq  zmovlp          ; jump if e
4d9a 20e94d 908:          jsr  relget          ; get low b
4d9d 18 909:          clc
4d9e 6d054e 910:          adc  zoffset
4da1 b1d7 911:          lda  (zwptr),y
4da3 6d064e 912:          adc  zoffset+1
4da6 91d7 913:          sta  (zwptr),y
4da8 4c954d 914:          jmp  zhbytlp
4dab 915:
4dab ae044e 916: zmovlp:   ldx  segment          ; g
4dae bd4f4f 917:          lda  segtab+blkadr,x
4db1 8dcc4d 918:          sta  zmovfr+1
4db4 bd504f 919:          lda  segtab+blkadr+1,x
4db7 8dcd4d 920:          sta  zmovfr+2
4dba bd534f 921:          lda  segtab+blkdes,x
4dbd 8dcf4d 922:          sta  zmovto+1
4dc0 bd544f 923:          lda  segtab+blkdes+1,x
4dc3 8dd04d 924:          sta  zmovto+2
4dc6 925:          ; get # pag
4dc6 bc524f 926:          ldy  segtab+blkbyt+1,x
4dc9 a200 927:          ldx  #0
4dcb bdf5ff 928: zmovfr:   lda  $ffff,x
4dce 9df5ff 929: zmovto:   sta  $ffff,x
4dd1 e8 930:          inx
4dd2 d0f7 931:          bne  zmovfr
4dd4 eecd4d 932:          inc  zmovfr+2
4dd7 eed04d 933:          inc  zmovto+2
4dda 88 934:          dey
4ddb 10ee 935:          bpl  zmovfr          ; move even
4ddd 936:          ; bytes mo
4ddd ad044e 937:          lda  segment
4de0 18 938:          clc
4de1 690c 939:          adc  #seglen          ; go to nex
4de3 8d044e 940:          sta  segment
4de6 4c454d 941:          jmp  segloop
4de9 942:
4de9 adffff 943: relget:   lda  $ffff
4dec eeea4d 944:          inc  relget+1
4def d003 945:          bne  nc1
4df1 eeeb4d 946:          inc  relget+2
4df4 60 947: nc1:      rts
4df5 948:
4df5 20e94d 949: getzwp:   jsr  relget          ; t
4df8 85d7 950:          sta  zwptr          ; save addr

```

```

4dfa 20e94d 951:      jsr  relget
4dfd a000 952:      ldy  #0
4dff 85d8 953:      sta  zwptr+1
4e01 05d7 954:      ora  zwptr      ; check if
4e03 60 955:      rts
4e04 956:
4e04 00 957: segment:  dc.b  0      ; current s
4e05 0000 958: zoffset:  dc.w  0      ; offset to
4e07 959: ;
4e07 960: ;~~~~~
4e07 961: ;
4e07 962: ;      Relocation data table
4e07 963: ;      -----
4e07 964: ;      We resolve Word locations First!
4e07 965: ;
4e07 966: ;      Things like  Lda $5000
4e07 967: ;      Sta $5000
4e07 968: ;      lda $5000,y
4e07 969: ;      jsr $5000
4e07 970: ;      jmp $5000
4e07 971: ;
4e07 972: ;      Well you get the Idea..
4e07 973: ;
4e07 974: ;~~~~~
4e07 975: ;
4e07 976: rtable:
4e07 d143 977:      dc.w  w01+1
4e09 d643 978:      dc.w  w02+1
4e0b db43 979:      dc.w  w03+1
4e0d de43 980:      dc.w  w04+1
4e0f e343 981:      dc.w  w05+1
4e11 e643 982:      dc.w  w06+1
4e13 f143 983:      dc.w  w07+1
4e15 984: ;      dc.w  w08+1
4e15 4544 985:      dc.w  w09
4e17 986:
4e17 4744 987:      dc.w  w10
4e19 4944 988:      dc.w  w11
4e1b 4b44 989:      dc.w  w12
4e1d 4d44 990:      dc.w  w13
4e1f 4f44 991:      dc.w  w14
4e21 5244 992:      dc.w  w15+1
4e23 5744 993:      dc.w  w16+1
4e25 5a44 994:      dc.w  w17+1
4e27 7744 995:      dc.w  w18+1
4e29 7a44 996:      dc.w  W19+1
4e2b 997:
4e2b 7d44 998:      dc.w  w20+1
4e2d 8d44 999:      dc.w  w21+1
4e2f 9044 1000:      dc.w  w22+1
4e31 ad44 1001:      dc.w  w23+1
4e33 b044 1002:      dc.w  w24+1
4e35 b344 1003:      dc.w  w25+1

```

4e37	b844	1004:	dc.w	w26+1
4e39	bb44	1005:	dc.w	w27+1
4e3b	c144	1006:	dc.w	w28+1
4e3d	c644	1007:	dc.w	w29+1
4e3f		1008:		
4e3f	cd44	1009:	dc.w	w30+1
4e41	df44	1010:	dc.w	w31+1
4e43	e644	1011:	dc.w	w32+1
4e45	ee44	1012:	dc.w	w33+1
4e47	f444	1013:	dc.w	w34+1
4e49	f944	1014:	dc.w	w35+1
4e4b	ff44	1015:	dc.w	w36+1
4e4d	0a45	1016:	dc.w	w37+1
4e4f	1b45	1017:	dc.w	w38+1
4e51	2045	1018:	dc.w	w39+1
4e53		1019:		
4e53	2645	1020:	dc.w	w40+1
4e55	2a45	1021:	dc.w	w41+1
4e57	5145	1022:	dc.w	w42+1
4e59	5645	1023:	dc.w	w43+1
4e5b	6a45	1024:	dc.w	w44+1
4e5d	6d45	1025:	dc.w	w45+1
4e5f	8645	1026:	dc.w	w46+1
4e61	8d45	1027:	dc.w	w47+1
4e63	9245	1028:	dc.w	w48+1
4e65	9545	1029:	dc.w	w49+1
4e67		1030:		
4e67	9a45	1031:	dc.w	w50+1
4e69	9d45	1032:	dc.w	w51+1
4e6b	a845	1033:	dc.w	w52+1
4e6d	ad45	1034:	dc.w	w53+1
4e6f	b045	1035:	dc.w	w54+1
4e71	b645	1036:	dc.w	w55+1
4e73	c145	1037:	dc.w	w56+1
4e75	ca45	1038:	dc.w	w57+1
4e77	cd45	1039:	dc.w	w58+1
4e79	d845	1040:	dc.w	w59+1
4e7b		1041:		
4e7b	e545	1042:	dc.w	w60+1
4e7d	fa45	1043:	dc.w	w61+1
4e7f	fd45	1044:	dc.w	w62+1
4e81	0246	1045:	dc.w	w63+1
4e83	0746	1046:	dc.w	w64+1
4e85	1346	1047:	dc.w	w65+1
4e87	1646	1048:	dc.w	w66+1
4e89	1f46	1049:	dc.w	w67+1
4e8b	2646	1050:	dc.w	w68+1
4e8d	2946	1051:	dc.w	w69+1
4e8f		1052:		
4e8f	3046	1053:	dc.w	w70+1
4e91	3346	1054:	dc.w	w71+1
4e93	3846	1055:	dc.w	w72+1
4e95	4946	1056:	dc.w	w73+1

```

4e97 4e46 1057:      dc.w  w74+1
4e99 5146 1058:      dc.w  w75+1
4e9b 5646 1059:      dc.w  w76+1
4e9d 5946 1060:      dc.w  w77+1
4e9f 5c46 1061:      dc.w  w78+1
4ea1 5f46 1062:      dc.w  w79+1
4ea3      1063:
4ea3 6646 1064:      dc.w  w80+1
4ea5 6946 1065:      dc.w  w81+1
4ea7 6e46 1066:      dc.w  w82+1
4ea9 8146 1067:      dc.w  w83+1
4eab 8a46 1068:      dc.w  w84+1
4ead 9746 1069:      dc.w  w85+1
4eaf a046 1070:      dc.w  w86+1
4eb1 a346 1071:      dc.w  w87+1
4eb3 a646 1072:      dc.w  w88+1
4eb5 ab46 1073:      dc.w  w89+1
4eb7 b046 1074:      dc.w  w90+1
4eb9      1075:
4eb9 b346 1076:      dc.w  w91+1
4ebb b846 1077:      dc.w  w92+1
4ebd c346 1078:      dc.w  w93+1
4ebf c646 1079:      dc.w  w94+1
4ec1 cb46 1080:      dc.w  w95+1
4ec3 ce46 1081:      dc.w  w96+1
4ec5 e146 1082:      dc.w  w97+1
4ec7 e646 1083:      dc.w  w98+1
4ec9 e946 1084:      dc.w  w99+1
4ecb ef46 1085:      dc.w  w100+1
4ecd      1086:
4ecd f446 1087:      dc.w  w101+1
4ecf f746 1088:      dc.w  w102+1
4ed1 fd46 1089:      dc.w  w103+1
4ed3 0847 1090:      dc.w  w104+1
4ed5 0b47 1091:      dc.w  w105+1
4ed7 1147 1092:      dc.w  w106+1
4ed9 1447 1093:      dc.w  w107+1
4edb 1947 1094:      dc.w  w108+1
4edd 1c47 1095:      dc.w  w109+1
4edf 1f47 1096:      dc.w  w110+1
4ee1      1097:
4ee1 2747 1098:      dc.w  w111+1
4ee3 2a47 1099:      dc.w  w112+1
4ee5 2d47 1100:      dc.w  w113+1
4ee7 3047 1101:      dc.w  w114+1
4ee9 3347 1102:      dc.w  w115+1
4eeb 3647 1103:      dc.w  w116+1
4eed 3947 1104:      dc.w  w117+1
4eef 3c47 1105:      dc.w  w118+1
4ef1 3f47 1106:      dc.w  w119+1
4ef3      1107:
4ef3 4247 1108:      dc.w  w120+1
4ef5 4547 1109:      dc.w  w121+1

```

```

4ef7 4a47 1110:      dc.w  w122+1
4ef9 4f47 1111:      dc.w  w123+1
4efb 5247 1112:      dc.w  w124+1
4efd 5847 1113:      dc.w  w125+1
4eff 5b47 1114:      dc.w  w126+1
4f01 5e47 1115:      dc.w  w127+1
4f03 6147 1116:      dc.w  w128+1
4f05 6447 1117:      dc.w  w129+1
4f07      1118:
4f07 6a47 1119:      dc.w  w130+1
4f09 6d47 1120:      dc.w  w131+1
4f0b 7247 1121:      dc.w  w132+1
4f0d 7747 1122:      dc.w  w133+1
4f0f 8847 1123:      dc.w  w134+1
4f11 a447 1124:      dc.w  w135+1
4f13 ae47 1125:      dc.w  w136+1
4f15 b147 1126:      dc.w  w137+1
4f17 b947 1127:      dc.w  w138+1
4f19 bc47 1128:      dc.w  w139+1
4f1b      1129:
4f1b bf47 1130:      dc.w  w140+1
4f1d c747 1131:      dc.w  w141+1
4f1f      1132:
4f1f 0000 1133:      dc.w  0          ; ok done w
4f21      1134: ;~~~~~
4f21      1135: ;
4f21      1136: ;Ok these are low Byte Locations.. only one byte changed!
4f21      1137: ; OK Resolving Low Byte's First
4f21      1138: ;
4f21 e943 1139:      dc.w  lb01+1
4f23 2944 1140:      dc.w  lb02+1
4f25 3344 1141:      dc.w  lb03+1
4f27 3b44 1142:      dc.w  lb04+1
4f29 0d45 1143:      dc.w  lb05+1
4f2b d045 1144:      dc.w  lb06+1
4f2d      1145:
4f2d 7047 1146:      dc.w  lb07+1
4f2f 0000 1147:      dc.w  0          ; ok done w
4f31      1148:
4f31      1149: ;~~~~~
4f31      1150: ;Now to resolve High Bytes..... Only one byte to change!
4f31      1151: ; Here is what is needed... location of high byte in .wor
4f31      1152: ; Second thing needed is the low byte of the address you a
4f31      1153: ;to adjust!
4f31      1154: ;
4f31      1155: ;
4f31 ed43 1156:      dc.w  hb01+1
4f33 14    1157:      dc.b  l25d0
4f34      1158:
4f34 2e44 1159:      dc.w  Hb02+1
4f36 45    1160:      dc.b  g_handler
4f37      1161:
4f37 3744 1162:      dc.w  hb03+1

```

```

4f39 54 1163:      dc.b 12610
4f3a      1164:
4f3a 4044 1165:      dc.w hb04+1
4f3c 40 1166:      dc.b 12Efc
4f3d      1167:
4f3d 1145 1168:      dc.w hb05+1
4f3f fc 1169:      dc.b 129b8
4f40      1170:
4f40 d445 1171:      dc.w Hb06+1
4f42 3c 1172:      dc.b 129f8
4f43      1173:
4f43 7547 1174:      dc.w hb07+1
4f45 40 1175:      dc.b 129fc
4f46      1176:
4f46 0000 1177:      dc.w 0          ; ok we are
4f48 00 1178:      dc.b 0          ; ok we are
4f49      1179:
4f49      1180: ;~~~~~
4f49      1181: ;End of relocation Table!!!
4f49      1182: ;
4f49      1183: ;
4f49      1184: segtab:
4f49 074e 1185:      dc.w rtable      ; address of reloca
4f4b 1444 1186:      dc.w initz      ; assembly start ad
4f4d 0000 1187:      dc.w 0          ; relocate to addre
4f4f 1444 1188:      dc.w initz      ; absolute start ad
4f51 2c09 1189:      dc.w zend-initz  ; size of block
4f53 0000 1190:      dc.w 0          ; destination of bl
4f55 0000 1191:      dc.w 0          ; END of segments
4f57      1192:
4f57      1193: ;
4f57      1194: ;
4f57      1195: ;
4f57      1196: ;
4f57      1197: ;
4f57      1198: ;
4f57      1199: help_menu:
4f57      1200:      .include      notice
5435      1201:
5435 60 1202:      rts
5436      1203: ;
5436      1204: ;~~~~~
5436      1205: ; GET KEY ROUTEEN
5436      1206: ;this routeen get's one key and return it to the accumlato
5436      1207: ;~~~~~
5436      1208: ;
5436 86b6 1209: get_key:      stx  $b6
5438 84b7 1210:      sty  $b7
543a 204254 1211:      jsr  .a
543d a6b6 1212:      ldx  $b6
543f a4b7 1213:      ldy  $b7
5441 60 1214:      rts
5442 ad25e4 1215: .a:      lda  $e420+5

```



```

5445 48 1216:          pha
5446 ad24e4 1217:      lda  $e420+4
5449 48 1218:          pha
544a 60 1219:          rts
544b      1220:
544b      1221:
544b      1222: ;
544b      1223: ver_num:
544b 48 1224:          pha
544c 4a 1225:          lsr
544d 4a 1226:          lsr
544e 4a 1227:          lsr
544f 4a 1228:          lsr
5450 205b54 1229:      jsr  .a
5453 a92e 1230:      lda  #'.'
5455 20a454 1231:      jsr  echo
5458 68 1232:          pla
5459 290f 1233:      and  #$0f
545b c90a 1234: .a:      cmp  #$0a
545d b004 1235:          bcs  .b
545f 0930 1236:          ora  #$30
5461 d002 1237:          bne  .c
5463 6936 1238: .b:      adc  #$36
5465 20a454 1239: .c:      jsr  echo
5468 60 1240:          rts
5469      1241:
5469      1242:
5469      1243: ;      print the byte currently in Accum.
5469      1244: ;      ~~~~~
5469      1245:
5469 a200 1246: pr_byte:      ldx  #0
546b 86d5 1247: pr_card:      stx  fr0+1
546d      1248:
546d 85d4 1249: pr_num:      sta  fr0
546f 20aad9 1250:          jsr  ifp          ;fl
5472 20e6d8 1251:          jsr  fasc          ;to
5475 a000 1252:          ldy  #0
5477 b1f3 1253: .prbloop:      lda  (inbuff),y
5479 08 1254:          php
547a 297f 1255:          and  #$7f          ;la
547c 20a454 1256:          jsr  echo
547f c8 1257:          iny
5480 28 1258:          plp
5481 10f4 1259:          bpl  .prbloop
5483 60 1260:          rts
5484      1261:
5484      1262: ;
5484      1263:
5484      1264: ; prints: prints string with address in PR_PTR, length by
5484      1265: ; ~~~~~
5484      1266:
5484      1267: prints:
5484      1268: echos:

```

```

5484      1269: pri2:
5484 a000 1270:      ldy  #0
5486 b188 1271:      lda  (pr_ptr),y      ;ge
5488 f017 1272:      beq  .xpri      ;ex
548a 8da354 1273:      sta  .maxlth
548d c8 1274:      iny
548e 8ca254 1275:      sty  .tempy
5491      1276:
5491 aca254 1277: .pril:      ldy  .tempy
5494 b188 1278:      lda  (pr_ptr),y
5496 20a454 1279:      jsr  echo
5499 eea254 1280: .prnext:      inc  .tempy
549c cea354 1281:      dec  .maxlth
549f d0f0 1282:      bne  .pril
54a1 60 1283: .xpri:      rts
54a2      1284:
54a2 01 1285: .tempy:      dc.b 1
54a3 01 1286: .maxlth:      dc.b 1
54a4      1287:
54a4      1288: ;
54a4      1289: ;-----
54a4      1290: ; PRINT BYTE
54a4      1291: ;-----
54a4      1292: echo:
54a4 85b8 1293: sprint:      sta  $b8
54a6 a904 1294:      lda  #$04
54a8 8d4a03 1295:      sta  $034a
54ab 86b6 1296:      stx  $b6
54ad 84b7 1297:      sty  $b7
54af a206 1298:      ldx  #$06
54b1 bd01e4 1299:      lda  $e400+1,x
54b4 85b3 1300:      sta  $b3
54b6 bd00e4 1301:      lda  $e400,x
54b9 85b2 1302:      sta  $b2
54bb a5b8 1303:      lda  $b8
54bd 20cc54 1304:      jsr  .a
54c0 a6b6 1305:      ldx  $b6
54c2 a4b7 1306:      ldy  $b7
54c4 a90c 1307:      lda  #$0c
54c6 8d4a03 1308:      sta  $034a
54c9 a5b8 1309:      lda  $b8
54cb 60 1310:      rts
54cc      1311: ;
54cc e6b2 1312: .a:      inc  $b2
54ce 6cb200 1313:      jmp  ($b2)
54d1 60 1314:      rts
54d2      1315:
54d2      1316:
54d2      1317:
54d2      1318:
54d2      1319: ;
54d2      1320: ;-----
54d2      1321: ; jsr printsi

```

```

54d2      1322: ;this prints an inline string
54d2      1323: ;   terminated by $ff
54d2      1324: ;-----
54d2      1325: ;   jsr prints
54d2      1326: echosi:
54d2 68    1327: printsi:      pla
54d3 8de354 1328:              sta  pstr+1
54d6 68    1329:              pla
54d7 8de454 1330:              sta  pstr+2
54da eee354 1331: prsl:        inc  pstr+1
54dd d003   1332:              bne  pstr
54df eee454 1333:              inc  pstr+2
54e2 adffff 1334: pstr:        lda  $ffff
54e5 c9ff   1335:              cmp  #$ff
54e7 f006   1336:              beq  estri
54e9 202855 1337:              jsr  fast_output
54ec 4cda54 1338:              jmp  prsl
54ef ade454 1339: estri:       lda  pstr+2
54f2 48     1340:              pha
54f3 ade354 1341:              lda  pstr+1
54f6 48     1342:              pha
54f7 60     1343:              rts
54f8      1344: ;
54f8      1345: ;-----
54f8      1346: ;actual screen handler
54f8      1347: ;screen offset definitions
54f8      1348: ;-----
54f8 0000   1349: scr_offset:   dc.w  0
54fa 2800   1350:              dc.w  40
54fc 5000   1351:              dc.w  80
54fe 7800   1352:              dc.w 120
5500 a000   1353:              dc.w 160
5502 c800   1354:              dc.w 200
5504 f000   1355:              dc.w 240
5506 1801   1356:              dc.w 280
5508 4001   1357:              dc.w 320
550a 6801   1358:              dc.w 360
550c 9001   1359:              dc.w 400
550e b801   1360:              dc.w 440
5510 e001   1361:              dc.w 480
5512 0802   1362:              dc.w 520
5514 3002   1363:              dc.w 560
5516 5802   1364:              dc.w 600
5518 8002   1365:              dc.w 640
551a a802   1366:              dc.w 680
551c d002   1367:              dc.w 720
551e f802   1368:              dc.w 760
5520 2003   1369:              dc.w 800
5522 4803   1370:              dc.w 840
5524 7003   1371:              dc.w 880
5526 9803   1372:              dc.w 920
5528      1373: ;
5528      1374: fast_output:

```

```

5528 855a 1375:      sta  $5a
552a ad0407 1376:      lda  sc.redirect
552d c902 1377:      cmp  #print_p
552f f01f 1378:      beq  .go_cio
5531      1379: .leo:
5531 adff02 1380:      lda  $02ff      ; CONTROL 1
5534 c900 1381:      cmp  #$00
5536 d0f9 1382:      bne  .leo
5538      1383:
5538 a55a 1384:      lda  $5a
553a c920 1385:      cmp  #32
553c 9004 1386:      bcc  .do_look
553e c97d 1387:      cmp  #125
5540 9013 1388:      bcc  .do_here
5542 a000 1389: .do_look:      ldy  #0
5544 b94b56 1390: .lookup:      lda  .char_table,y
5547 f00c 1391:      beq  .do_here
5549 c55a 1392:      cmp  $5a
554b f003 1393:      beq  .go_cio
554d c8 1394:      iny
554e d0f4 1395:      bne  .lookup
5550      1396:
5550      1397:
5550 a55a 1398: .go_cio:      lda  $5a
5552 4c5b56 1399:      jmp  putlocal
5555      1400:
5555      1401:
5555 a000 1402: .do_here:      ldy  #0
5557 a55d 1403:      lda  $5d
5559 915e 1404:      sta  ($5e),y
555b a55a 1405:      lda  $5a
555d c99b 1406:      cmp  #$9b
555f f044 1407:      beq  .docr
5561 202a56 1408: .notcr:      jsr  .get_adr
5564 a55a 1409:      lda  $5a
5566 297f 1410:      and  #$7f
5568 c920 1411:      cmp  #32
556a 9007 1412:      bcc  .add64
556c c960 1413:      cmp  #96
556e 9009 1414:      bcc  .sub32
5570 4c7c55 1415:      jmp  .asis
5573 18 1416: .add64:      clc
5574 6940 1417:      adc  #64
5576 4c7c55 1418:      jmp  .asis
5579 38 1419: .sub32:      sec
557a e920 1420:      sbc  #32
557c 245a 1421: .asis:      bit  $5a
557e 1002 1422:      bpl  .xxlate
5580 0980 1423:      ora  #$80
5582 a000 1424: .xxlate:      ldy  #0
5584 915e 1425:      sta  ($5e),y
5586 e655 1426:      inc  $55
5588 e663 1427:      inc  $63

```

```

558a a555 1428:      lda  $55
558c c928 1429:      cmp  #40
558e b019 1430:      bcs  .next_row
5590 c8 1431:      iny
5591 b15e 1432:      lda  ($5e),y
5593 855d 1433:      sta  $5d
5595 0980 1434:      ora  #$80
5597 aef002 1435:     ldx  752
559a d002 1436:      bne  .nocurs1
559c 915e 1437:      sta  ($5e),y
559e e65e 1438: .nocurs1:      inc  $5e
55a0 d002 1439:      bne  .nooav
55a2 e65f 1440:      inc  $5e+1
55a4 60 1441: .nooav:      rts
55a5 a900 1442: .docr:      lda  #0
55a7 8563 1443:      sta  $63
55a9 a552 1444: .next_row:      lda  $52
55ab 8555 1445:      sta  $55
55ad e654 1446:      inc  $54
55af a454 1447:      ldy  $54
55b1 c018 1448:      cpy  #24
55b3 b013 1449:      bcs  .scroll
55b5 202a56 1450:     jsr  .get_adr
55b8 a000 1451:      ldy  #0
55ba b15e 1452:      lda  ($5e),y
55bc 855d 1453:      sta  $5d
55be aef002 1454:     ldx  752
55c1 d004 1455:      bne  .nocurs2
55c3 0980 1456:      ora  #$80
55c5 915e 1457:      sta  ($5e),y
55c7 60 1458: .nocurs2:      rts
55c8 1459:
55c8 c654 1460: .scroll:      dec  $54
55ca a558 1461:      lda  $58
55cc 8568 1462:      sta  $68
55ce 18 1463:      clc
55cf 6928 1464:      adc  #40
55d1 855e 1465:      sta  $5e
55d3 a559 1466:      lda  $58+1
55d5 8569 1467:      sta  $68+1
55d7 6900 1468:      adc  #0
55d9 855f 1469:      sta  $5e+1
55db a000 1470:      ldy  #0
55dd a203 1471:      ldx  #3
55df b15e 1472: .sloop:      lda  ($5e),y
55e1 9168 1473:      sta  ($68),y
55e3 c8 1474:      iny
55e4 d0f9 1475:      bne  .sloop
55e6 e65f 1476:      inc  $5e+1
55e8 e669 1477:      inc  $68+1
55ea ca 1478:      dex
55eb d0f2 1479:      bne  .sloop
55ed b15e 1480: .sloop2:      lda  ($5e),y

```

```

55ef 9168 1481:      sta  ($68),y
55f1 c8 1482:      iny
55f2 c098 1483:      cpy  #152
55f4 90f7 1484:      bcc  .scloop2
55f6 a558 1485:      lda  $58
55f8 18 1486:      clc
55f9 6998 1487:      adc  #920&$ff
55fb 855e 1488:      sta  $5e
55fd a559 1489:      lda  $58+1
55ff 6903 1490:      adc  #920/256
5601 855f 1491:      sta  $5e+1
5603 a000 1492:      ldy  #0
5605 98 1493:      tya
5606 915e 1494: .clear:      sta  ($5e),y
5608 c8 1495:      iny
5609 c028 1496:      cpy  #40
560b 90f9 1497:      bcc  .clear
560d aef002 1498:      ldx  752
5610 d006 1499:      bne  .nocurs3
5612 a980 1500:      lda  #$80
5614 a455 1501:      ldy  $55
5616 915e 1502:      sta  ($5e),y
5618 a900 1503: .nocurs3:      lda  #0
561a 855d 1504:      sta  $5d
561c a55e 1505:      lda  $5e
561e 18 1506:      clc
561f 6555 1507:      adc  $55
5621 855e 1508:      sta  $5e
5623 a55f 1509:      lda  $5e+1
5625 6900 1510:      adc  #0
5627 855f 1511:      sta  $5e+1
5629 60 1512:      rts
562a a554 1513: .get_adr:      lda  $54
562c 0a 1514:      asl
562d a8 1515:      tay
562e a558 1516:      lda  $58
5630 18 1517:      clc
5631 79f854 1518:      adc  scr_offset,y
5634 855e 1519:      sta  $5e
5636 a559 1520:      lda  $58+1
5638 79f954 1521:      adc  scr_offset+1,y
563b 855f 1522:      sta  $5e+1
563d a55e 1523:      lda  $5e
563f 18 1524:      clc
5640 6555 1525:      adc  $55
5642 855e 1526:      sta  $5e
5644 a55f 1527:      lda  $5e+1
5646 6900 1528:      adc  #0
5648 855f 1529:      sta  $5e+1
564a 60 1530:      rts
564b 1531: .char_table:
564b 1b1c1d 1532:      dc.b  27,28,29,30,31,125,126,127
5653 9c9d9e 1533:      dc.b  156,157,158,159,253,254,255

```

```

565b 1534:
565b 1535:
565b 1536: putlocal:
565b a20b 1537:      ldx  #11
565d 8e4203 1538:      stx  $0342
5660 a200 1539:      ldx  #0
5662 8e4803 1540:      stx  $0348
5665 8e4903 1541:      stx  $0349
5668 4c56e4 1542:      jmp  $e456
566b 1543: ;
566b 1544:
566b 1545:
566b 1546:
566b 1547:
566b 1548: ;~~~~~
566b 1549: ; OS on and off ROUTEEN
566b 1550: ;~~~~~
566b 1551: ;
566b 1552:
566b 1553: os_off:
566b 78 1554:      SEI
566c ad01d3 1555:      LDA  PORTB
566f 29fe 1556:      AND  #$FE
5671 8d01d3 1557:      STA  PORTB
5674 58 1558:      CLI
5675 60 1559:      RTS
5676 1560: ;
5676 1561: os_on:
5676 78 1562:      SEI
5677 ad01d3 1563:      LDA  PORTB
567a 0901 1564:      ORA  #1
567c 8d01d3 1565:      STA  PORTB
567f 58 1566:      CLI
5680 60 1567:      RTS
5681 1568: ;~~~~~
5681 1569: ;checks user security level pointed to by S_PTR to see if
5681 1570: ;bit specified in the A register is ON. Returns MINus if
5681 1571: ;user doesn't have the specified security level.
5681 1572: ;~~~~~
5681 1573:
5681 8cdc56 1574: seclvl:      sty  .tempy
5684 8edd56 1575:      stx  .tempx
5687 c900 1576:      cmp  #0      ;se
5689 f042 1577:      beq  .passed  ;ye
568b a8 1578:      tay      ;-1
568c a980 1579:      lda  #128
568e 8dde56 1580:      sta  .temp      ;in
5691 a900 1581:      lda  #0
5693 8ddf56 1582:      sta  .temp+1
5696 8de056 1583:      sta  .temp+2
5699 8de156 1584:      sta  .temp+3
569c 88 1585:      dey
569d f011 1586:      beq  .mask      ;if

```

```

569f 4ede56 1587: .shloop:      lsr  .temp          ;sh
56a2 6edf56 1588:                ror  .temp+1
56a5 6ee056 1589:                ror  .temp+2
56a8 6ee156 1590:                ror  .temp+3
56ab 88    1591:                dey
56ac d0f1   1592:                bne  .shloop
56ae a000   1593:                ldy  #0
56b0 b9de56 1594: .mask:         lda  .temp,y          ;no
56b3 3182   1595:                and  (s_ptr),y
56b5 99de56 1596:                sta  .temp,y
56b8 c8     1597:                iny
56b9 c004   1598:                cpy  #4
56bb d0f3   1599:                bne  .mask
56bd adde56 1600: .check:        lda  .temp
56c0 0ddf56 1601:                ora  .temp+1          ; i
56c3 0de056 1602:                ora  .temp+2
56c6 0de156 1603:                ora  .temp+3
56c9 c900   1604:                cmp  #0
56cb f004   1605:                beq  .failed
56cd a900   1606: .passed:       lda  #0
56cf f002   1607:                beq  .return
56d1 a980   1608: .failed:       lda  #$80
56d3 08     1609: .return:       php
56d4 acdc56 1610:                ldy  .tempy
56d7 aedd56 1611:                ldx  .tempx
56da 28     1612:                plp
56db 60     1613:                rts
56dc       1614:
56dc 00     1615: .tempy:        dc.b  0
56dd 00     1616: .tempx:        dc.b  0
56de 000000 1617: .temp:         dc.b  0,0,0,0
56e2       1618:
56e2       1619:
56e2       1620: ; sets the bit specified (in the A register) of the se
56e2       1621: ; level pointed to by S_PTR to a TRUE value.
56e2       1622: ; ~~~~~
56e2       1623:
56e2 8c1e57 1624: setlvl:        sty  .tempy
56e5 8e1f57 1625:                stx  .tempx
56e8 c900   1626:                cmp  #0              ;se
56ea f031   1627:                beq  .return          ;ye
56ec a8     1628:                tay              ;-
56ed a980   1629:                lda  #128
56ef 8d2057 1630:                sta  .temp          ;in
56f2 a900   1631:                lda  #0
56f4 8d2157 1632:                sta  .temp+1
56f7 8d2257 1633:                sta  .temp+2
56fa 8d2357 1634:                sta  .temp+3
56fd 88     1635:                dey
56fe f011   1636:                beq  .mask          ;if
5700 4e2057 1637: .shloop:       lsr  .temp          ;sh
5703 6e2157 1638:                ror  .temp+1
5706 6e2257 1639:                ror  .temp+2

```



```

5709 6e2357 1640:      ror  .temp+3
570c 88 1641:      dey
570d d0f1 1642:      bne  .shloop
570f a000 1643:      ldy  #0
5711 b92057 1644: .mask:      lda  .temp,y      ;no
5714 1182 1645:      ora  (s_ptr),y
5716 9182 1646:      sta  (s_ptr),y
5718 c8 1647:      iny
5719 c004 1648:      cpy  #4
571b d0f4 1649:      bne  .mask
571d 60 1650: .return:      rts
571e 1651:
571e 00 1652: .tempy:      dc.b  0
571f 00 1653: .tempx:      dc.b  0
5720 000000 1654: .temp:      dc.b  0,0,0,0
5724 1655:
5724 1656: ; ~~~~~
5724 1657: set_g_handler:
5724 206b56 1658:      jsr  os_off
5727 1659:      mea  tsr,s_ptr
5727 a9f5 ----:      lda  #low tsr
5729 8582 ----:      sta  s_ptr
572b a9ff ----:      lda  #high tsr
572d 8583 ----:      sta  s_ptr + 1
572f a91a 1660:      lda  #gdevice
5731 20e256 1661:      jsr  setlvl
5734 207656 1662:      jsr  os_on
5737 60 1663:      rts
5738 1664:
5738 1665:
5738 1666: win_end:      ds.b  0
5738 1667: ;

```

End assembly: no errors