


```

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: ;      Peek.com
0000 9: ;
0000 10: ; This is a program to peak a byte and display it to
0000 11: ;      init 2/24/2010   sjc new design
0000 12: ;      added hex delimiter      2/25/2010
0000 13: ;      added new code for registered user 2/26/201
0000 14: ;
0000 15: ;Updated the screen Print Utility so it will work with the
0000 16: ; This change will also help "Print P:"      8/1
0000 17: ;
0000 18: ;
0000 19: ;Also added code for Control+1 pause of screen to the high
0000 20: ;handling.      8/24/2012   sjc
0000 21: ;
0000 22: ;
0000 23: ;
0000 24: ;
0000 25: ; ~~~~~
0000 26: ;-----
0000 27: ; Notes: o This source code MAY NOT be placed for download
0000 28: ;      o "mea" is a macro that loads the address of the
0000 29: ;      into the pointer specified by the second field.
0000 30: ;-----
0000 31: ; Assembler: MADMAC (tm) ST Cross Assembler (Atari Corp)
0000 32: ;      XASM  st and IBM VERSIONS
0000 33: ;-----
0000 34:
0000 35: com_file_name:      .macro
0000 36: ;      dc.b 13,"filename.ext",$9b
0000 37: ;      dc.b 12,"Peek.com  ", $9b
0000 38: ;      .endm
0000 39:
0000 40: ;
0000 41: ;~~~~~
0000 42: ; File revision history. Version Number in hex
0000 43: ;
0000 44: file_ver:      equ $16
0000 45: ;~~~~~
0000 46: ; the Month in dec for the first time this revision
0000 47: ;was compiled
0000 48: ;
0000 49: c_month:      equ 8
0000 50: ;~~~~~
0000 51: ; the day in dec for the first time this revision
0000 52: ;was compiled
0000 53: c_day:      equ 24

```

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

```

0000    107: ; 0 = Undefined
0000    108: ; 1 = English
0000    109: ; 2 = German
0000    110: ;
0000    111: r..language:      equ    1
0000    112: ;~~~~~
0000    113:
0000    114:
0000    115:          .include    equates
0000    116:          .include    globals
0000    117:          .include    macros
0000    118:
4000    119:          .org $4000
4000    120:
4000    121: win_start:
4000    122: header_info:
4000    123:          .include    header
42a7    124: ;
42a7    125: peekabyte:
42a7 ad01d3 126:          lda    portb
42aa 8d7e44 127:          sta    under.the.os.data
42ad    128:
42ad a03f 129:          ldy    #LBUF
42af b10a 130: .a:          lda    (comtab),y
42b1 c920 131:          cmp    #' '
42b3 f01f 132:          beq    .gotspacechar
42b5 c99b 133:          cmp    #$9b
42b7 f012 134:          beq    .noparmerror
42b9    135:
42b9 c8 136:          iny
42ba a906 137:          lda    #consol_start
42bc cd1fd0 138:          cmp    consol
42bf f00d 139:          beq    .tellfilehistory
42c1 a907 140:          lda    #consol_any
42c3 cd1fd0 141:          cmp    consol
42c6 d009 142:          bne    .abort
42c8 4caf42 143:          jmp    .a
42cb    144:
42cb 4c1d45 145: .noparmerror:      jmp    noparmerror
42ce 4cd34a 146: .tellfilehistory:  jmp    my_file_notice
42d1 4c8244 147: .abort:            jmp    peek.a.byte.abort
42d4    148:
42d4    149:
42d4    150: .gotspacechar:
42d4 a200 151:          ldx    #$00
42d6 c8 152:          iny
42d7    153:
42d7 b10a 154: .af:          lda    (comtab),y
42d9 20aa48 155:          jsr    upcase
42dc c948 156:          cmp    #'H'
42de f01c 157:          beq    .go_show_help
42e0 c924 158:          cmp    #'$'
42e2 f01b 159:          beq    .gothex

```

```

42e4 c93f 160:      cmp    #'?'
42e6 f0e6 161:      beq    .tellfilehistory
42e8 c99b 162:      cmp    #$9b
42ea f055 163:      beq    .oknum
42ec 9dd146 164:      sta    mynumberstring,x
42ef c930 165:      cmp    #'0'
42f1 904e 166:      bcc    .oknum
42f3 c93a 167:      cmp    #'9'+1
42f5 b04a 168:      bcs    .oknum
42f7 e8 169:      inx
42f8 c8 170:      iny
42f9 4cd742 171:      jmp    .af
42fc      172:
42fc 4c8244 173: .go_show_help:  jmp    peek.a.byte.abort
42ff      174:
42ff c8 175: .gothex:      iny
4300 a980 176:      lda    #$80
4302 8d7c44 177:      sta    inhexmode
4305 b10a 178: .h:      lda    (comtab),y
4307 20aa48 179:      jsr    upcase
430a 9dd146 180:      sta    mynumberstring,x
430d c99b 181:      cmp    #$9b
430f f031 182:      beq    .eol
4311 c920 183:      cmp    #$20
4313 f015 184:      beq    .got.more.here
4315 c930 185:      cmp    #'0'
4317 900d 186:      bcc    .badchar
4319 c93a 187:      cmp    #'9'+1
431b 9008 188:      bcc    .charok
431d c941 189:      cmp    #'A'
431f 9005 190:      bcc    .badchar
4321 c947 191:      cmp    #'F'+1
4323 b01c 192:      bcs    .oknum
4325 e8 193: .charok:      inx
4326 c8 194: .badchar:      iny
4327 4c0543 195:      jmp    .h
432a      196:
432a c8 197: .got.more.here:  iny
432b b10a 198:      lda    (comtab),y
432d c92f 199:      cmp    #'/'
432f d010 200:      bne    .oknum
4331 c99b 201:      cmp    #$9b
4333 f00c 202:      beq    .oknum
4335 c8 203:      iny
4336 b10a 204:      lda    (comtab),y
4338 c955 205:      cmp    #'U'
433a d005 206:      bne    .oknum
433c a901 207:      lda    #$01
433e 8d7d44 208:      sta    under.the.os
4341      209:
4341      210: .oknum:
4341 e8 211:      inx
4342 a99b 212: .eol:      lda    #$9b

```

```

4344 9dd146 213: .goodhexeol:      sta  mynumberstring,x
4347      214:
4347      215:
4347 ad7c44 216:      lda  inhexmode
434a c900  217:      cmp  #$00
434c f012  218:      beq  indecmode
434e 204c48 219:      jsr  dohexconversion
4351 ae8848 220:      ldx  expr+1
4354 ad8748 221:      lda  expr
4357 8d8044 222:      sta  source_location
435a 8e8144 223:      stx  source_location+1
435d      224:
435d 4cf043 225:      jmp  hex_screen_print
4360      226:
4360      227:
4360      228:
4360      229: indecmode:
4360 20b345 230:      jsr  get_number_cli
4363 8d8044 231:      sta  source_location
4366 8e8144 232:      stx  source_location+1
4369      233:
4369      234: dec_screen_print:
4369 20e448 235:      jsr  printsi
436c 9b204c 236:      dc.b  "$9b," Location ",-1
4378 ad8044 237:      lda  source_location
437b ae8144 238:      ldx  source_location+1
437e 20d247 239:      jsr  pr_card
4381      240:
4381 20e448 241:      jsr  printsi
4384 203dff 242:      dc.b  "=", -1
4387      243:
4387 ad8044 244:      lda  source_location
438a 8daa43 245:      sta  location_byte+1
438d ad8144 246:      lda  source_location+1
4390 8dab43 247:      sta  location_byte+2
4393      248:
4393 ad8044 249:      lda  source_location
4396 8dd243 250:      sta  location_byte2+1
4399 ad8144 251:      lda  source_location+1
439c 8dd343 252:      sta  location_byte2+2
439f      253:
439f ad7d44 254:      lda  under.the.os
43a2 c900  255:      cmp  #$00
43a4 f003  256:      beq  .not_under
43a6 20c74a 257:      jsr  os_off
43a9      258: .not_under:
43a9      259:
43a9 adffff 260: location_byte:  lda  $ffff
43ac 8d7f44 261:      sta  tammy.byte
43af      262:
43af ad7d44 263:      lda  under.the.os
43b2 c900  264:      cmp  #$00
43b4 f003  265:      beq  .not_under

```

```

43b6 20c04a 266:      jsr  os_on
43b9      267: ..not_under:
43b9 ad7f44 268:      lda  tammy.byte
43bc      269:
43bc 20d047 270:      jsr  pr_byte
43bf 20e448 271:      jsr  printsi
43c2 202020 272:      dc.b  " ",-1
43c7      273:
43c7 ad7d44 274:      lda  under.the.os
43ca c900   275:      cmp  #$00
43cc f003   276:      beq  ..not_under
43ce 20c74a 277:      jsr  os_off
43d1      278: ..not_under:
43d1      279:
43d1      280:
43d1      281:
43d1 adffff 282: location_byte2:      lda  $ffff
43d4 8d7f44 283:      sta  tammy.byte
43d7      284:
43d7 ad7d44 285:      lda  under.the.os
43da c900   286:      cmp  #$00
43dc f003   287:      beq  ..not_under
43de 20c04a 288:      jsr  os_on
43e1      289: ..not_under:
43e1 ad7f44 290:      lda  tammy.byte
43e4 203a49 291:      jsr  echo
43e7 20e448 292:      jsr  printsi
43ea 27     293:      dc.b  ""
43eb 9b9b9b 294:      dc.b  $9b,$9b,$9b,-1
43ef      295:
43ef 60     296:      rts
43f0      297:
43f0      298:
43f0      299: hex_screen_print:
43f0      300:
43f0 20e448 301:      jsr  printsi
43f3 9b204c 302:      dc.b  $9b," Location $",-1
4400 ad8044 303:      lda  source_location
4403 ae8144 304:      ldx  source_location+1
4406 202448 305:      jsr  pr_hex_card
4409      306:
4409 20e448 307:      jsr  printsi
440c 203d24 308:      dc.b  "=$",-1
4410      309:
4410 ad8044 310:      lda  source_location
4413 8d3344 311:      sta  .location_byte+1
4416 ad8144 312:      lda  source_location+1
4419 8d3444 313:      sta  .location_byte+2
441c      314:
441c ad8044 315:      lda  source_location
441f 8d5b44 316:      sta  .location_byte2+1
4422 ad8144 317:      lda  source_location+1
4425 8d5c44 318:      sta  .location_byte2+2

```

```

4428      319:
4428 ad7d44 320:      lda    under.the.os
442b c900  321:      cmp    #$00
442d f003  322:      beq    .not_under
442f 20c74a 323:      jsr    os_off
4432      324: .not_under:
4432      325:
4432 adffff 326: .location_byte:      lda    $ffff
4435 8d7f44 327:      sta    tammy.byte
4438      328:
4438 ad7d44 329:      lda    under.the.os
443b c900  330:      cmp    #$00
443d f003  331:      beq    .n_under
443f 20c04a 332:      jsr    os_on
4442      333: .n_under:
4442 ad7f44 334:      lda    tammy.byte
4445 203348 335:      jsr    pr_hex_byte
4448 20e448 336:      jsr    printsi
444b 202020 337:      dc.b   " ",-1
4450 ad7d44 338:      lda    under.the.os
4453 c900  339:      cmp    #$00
4455 f003  340:      beq    .location_byte2
4457 20c74a 341:      jsr    os_off
445a      342:
445a adffff 343: .location_byte2:      lda    $ffff
445d 8d7f44 344:      sta    tammy.byte
4460      345:
4460 ad7d44 346:      lda    under.the.os
4463 c900  347:      cmp    #$00
4465 f003  348:      beq    .tn_under
4467 20c04a 349:      jsr    os_on
446a      350: .tn_under:
446a ad7f44 351:      lda    tammy.byte
446d      352:
446d      353:
446d 203a49 354:      jsr    echo
4470 20e448 355:      jsr    printsi
4473 27     356:      dc.b   ""
4474 9b9b9b 357:      dc.b   $9b,$9b,$9b,-1
4478      358:
4478 20c04a 359:      jsr    os_on
447b      360:
447b 60     361:      rts
447c      362: ;~~~~~
447c      363: ;this tells me what mode to be in decimal or hex
447c      364: ;
447c 00     365: inhexmode:      dc.b   0
447d      366: ;
447d      367: ;~~~~~
447d      368: ;OS reference bytes
447d 00     369: under.the.os:      dc.b   0
447e 00     370: under.the.os.data: dc.b   0
447f 00     371: tammy.byte:      dc.b   0

```



```
4480      372:
4480      373: ;~~~~~
4480      374: ;this the the location of the byte we are going to read.
4480      375: ;
4480 0000 376: source_location:    dc.w  0
4482      377: ;~~~~~
4482      378: ; Aborted the program with a consol key
4482      379: ;
4482      380: ;~~~~~
4482      381: peek.a.byte.abort:
4482 20b548 382:          jsr  bell
4485 20e448 383:          jsr  printsi
4488 9b9b 384:          dc.b  $9b,$9b
448a 20 385:          dc.b  " "
448b 919292 386:          dc.b  <+128>,"
44ad 9b 387:          dc.b  $9B
44ae 20 388:          dc.b  " "
44af fca0a0 389:          dc.b  <+128>,"| Peek a byte i
44d1 9b 390:          dc.b  $9B
44d2 20 391:          dc.b  " "
44d3 fca0d7 392:          dc.b  <+128>,"| Written by: Steph
44f5 9b 393:          dc.b  $9B
44f6 20 394:          dc.b  " "
44f7 9a9292 395:          dc.b  <+128>,"
```