


```

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: ;      Makedir.com
0000 9: ;
0000 10: ; This is a program makes recursive directors
0000 11: ; recursive dir maker 06/05/2010 sjc
0000 12: ;
0000 13: ;Updated the screen Print Utility so it will work with the
0000 14: ; This change will also help "Print P:" 8/1
0000 15: ;
0000 16: ;
0000 17: ;Also added code for Control+1 pause of screen to the high
0000 18: ;handling. 8/28/2012 sjc
0000 19: ;
0000 20: ;
0000 21: ;
0000 22: ; ~~~~~
0000 23: ;-----
0000 24: ; Notes: o This source code MAY NOT be placed for download
0000 25: ; o "mea" is a macro that loads the address of the
0000 26: ; into the pointer specified by the second field.
0000 27: ;-----
0000 28: ; Assembler: MADMAC (tm) ST Cross Assembler (Atari Corp)
0000 29: ; XASM st and IBM VERSIONS
0000 30: ;-----
0000 31:
0000 32: com_file_name: .macro
0000 33: ; dc.b 12,"filename.ext",$9b
0000 34: ; dc.b 12,"Makedir.com ",$9b
0000 35: ; .endm
0000 36:
0000 37: ;
0000 38: ;~~~~~
0000 39: ; File revision history. Version Number in hex
0000 40: ;
0000 41: file_ver: equ $13
0000 42: ;~~~~~
0000 43: ; the Month in dec for the first time this revision
0000 44: ;was compiled
0000 45: ;
0000 46: c_month: equ 8
0000 47: ;~~~~~
0000 48: ; the day in dec for the first time this revision
0000 49: ;was compiled
0000 50: c_day: equ 28
0000 51: ;~~~~~
0000 52: ; the year in dec for the first time this revision
0000 53: ;was compiled

```

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

```

0000    107: ;
0000    108: r.language:      equ    1
0000    109: ;~~~~~
0000    110:
0000    111:
0000    112:          .include    equates
0000    113:          .include    globals
0000    114:          .include    macros
0000    115:
4000    116:          .org $4000
4000    117:
4000    118: win_start:
4000    119: header_info:
4000    120:          .include    header
42a7    121:
42a7    122: tammy:
42a7 a210  123:          ldx    #disk1
42a9 20264c 124:          jsr    close
42ac a220  125:          ldx    #disk2
42ae 20264c 126:          jsr    close
42b1    127:
42b1 a50a  128:          lda    comtab
42b3 18    129:          CLC
42b4 6903  130:          adc    # low ZCRNAME
42b6 8d8743 131:          sta    cli.crunch+1
42b9 a50b  132:          lda    comtab+1
42bb 6900  133:          adc    # high ZCRNAME
42bd 8d8843 134:          sta    cli.crunch+2
42c0 208643 135:          jsr    cli.crunch
42c3 d003  136:          bne    .get_parm1
42c5 4c8943 137:          jmp    parm_errors
42c8    138:
42c8    139: .get_parm1:
42c8 a50a  140:          lda    comtab
42ca 18    141:          CLC
42cb 6921  142:          adc    # low COMFNAM
42cd 8582  143:          sta    s_ptr
42cf a50b  144:          lda    comtab+1
42d1 6900  145:          adc    # high COMFNAM
42d3 8583  146:          sta    s_ptr+1
42d5    147:          mea    parm_1,d_ptr
42d5 a968  ----:          lda    #low parm_1
42d7 8584  ----:          sta    d_ptr
42d9 a944  ----:          lda    #high parm_1
42db 8585  ----:          sta    d_ptr + 1
42dd 208447 148:          jsr    tostr
42e0    149:
42e0    150:          mea    fname,s_ptr
42e0 a9f8  ----:          lda    #low fname
42e2 8582  ----:          sta    s_ptr
42e4 a945  ----:          lda    #high fname
42e6 8583  ----:          sta    s_ptr + 1
42e8    151:          mea    parm_1,d_ptr

```

```

42e8 a968 ----:    lda  #low parm_1
42ea 8584 ----:    sta  d_ptr
42ec a944 ----:    lda  #high parm_1
42ee 8585 ----:    sta  d_ptr + 1
42f0 20ad47 152:    jsr  concat          ; a
42f3      153:
42f3      154:    mea  parm_1,s_ptr
42f3 a968 ----:    lda  #low parm_1
42f5 8582 ----:    sta  s_ptr
42f7 a944 ----:    lda  #high parm_1
42f9 8583 ----:    sta  s_ptr + 1
42fb      155:    mea  filename,d_ptr      ;mo
42fb a908 ----:    lda  #low filename
42fd 8584 ----:    sta  d_ptr
42ff a946 ----:    lda  #high filename
4301 8585 ----:    sta  d_ptr + 1
4303 205d47 156:    jsr  toascii
4306      157:
4306      158:
4306 20d946 159:    jsr  make_path
4309      160:
4309      161:
4309 ad1fd0 162:    lda  consol
430c c907  163:    cmp  #$07
430e f073  164:    beq  .no_tell          ;be
4310      165:
4310      166:
4310      167:
4310 200048 168:    jsr  printsi
4313 9b2074 169:    dc.b  $9b," this is my cli to par
4330 9b5061 170:    dc.b  $9b,"Parm_1 =",-1
433a      171:    mea  parm_1,pr_ptr
433a a968 ----:    lda  #low parm_1
433c 8588 ----:    sta  pr_ptr
433e a944 ----:    lda  #high parm_1
4340 8589 ----:    sta  pr_ptr + 1
4342 206048 172:    jsr  echos
4345      173:
4345 205a48 174:    jsr  cr_lf
4348      175:    mea  path1,pr_ptr
4348 a959 ----:    lda  #low path1
434a 8588 ----:    sta  pr_ptr
434c a946 ----:    lda  #high path1
434e 8589 ----:    sta  pr_ptr + 1
4350 206048 176:    jsr  echos
4353      177:
4353      178:
4353 200048 179:    jsr  printsi
4356 2023ff 180:    dc.b  " #",-1
4359 ad5846 181:    lda  totalpath
435c 202648 182:    jsr  pr_byte
435f      183:
435f 200048 184:    jsr  printsi

```

```

4362 9b9bff 185:          dc.b  $9b,$9b,-1
4365 204548 186:          jsr  get_key
4368 200048 187:          jsr  printsi
436b 20416e 188:          dc.b  " Any Key to Continue",$9b,
4382 60      189:          rts
4383        190:
4383        191: .no_tell:
4383 4c454a 192:          jmp  top
4386        193:
4386 4c0000 194: cli.crunch:      jmp  $0000
4389        195:
4389        196: parm_errors:
4389 a97d    197:          lda  #$7d
438b 20174a 198:          jsr  echo
438e 20a442 199:          jsr  tammy-3
4391        200:
4391 200048 201:          jsr  printsi
4394 9b9b   202:          dc.b  $9B,$9B
4396 20      203:          dc.b  " "
4397 d3f9ee  204:          dc.b  <+128>,"Syntax"
439d 20666f 205:          dc.b  " for use:",$9b
43a7 204d61 206:          dc.b  " Mkdir Dn:>Path",$9b
43b9 204d61 207:          dc.b  " Mkdir Dn:>real>dos>comm
43d7 204d61 208:          dc.b  " Mkdir Dx:>A>B>C>D>E>F>G
43fe 205468 209:          dc.b  " This Program will make Di
4422 202066 210:          dc.b  " for you up to the length
4445 202063 211:          dc.b  " command line buffer. 64
4465        212:
4465 9b      213:          dc.b  $9b
4466 ff     214:          dc.b  -1
4467 60      215:          rts
4468        216: ;
4468        217:
4468        218:
4468        219:
4468        220:
4468        221:
4468        222: parm_1:          ds.b  80
44b8        223: parm_2:          ds.b  80
4508        224: parm_3:          ds.b  80
4558        225: in_str:        ds.b  80          ;
45a8        226: instr:         ds.b  80          ;
45f8 0b3e77 227: fname:         dc.b  11,">wmsarp.sjc"
4604        228: ;
4604        229: my_count:      ds.b  1
4605        230:
4605 0000    231: tammy_c:       dc.w  0
4607 1a     232: stevel:        dc.b  26
4608        233:
4608        234:
4608        235: filename:      ds.b  80
4658        236:
4658        237:

```

```

4658 00    238: totalpath:      dc.b  0
4659      239: path1:          ds.b  128
46d9      240:
46d9      241:
46d9      242: ;~~~~~
46d9      243: ;Make Path
46d9      244: ;~~~~~
46d9      245: make_path:
46d9 a200  246:          ldx  #0
46db a001  247:          ldy  #1
46dd b96844 248: .a:      lda  parm_1,y
46e0 c93e  249:          cmp  #'>'
46e2 d006  250:          bne  .b
46e4 a900  251:          lda  #$00          ;'%
46e6 ee5846 252:          inc  totalpath
46e9 e8    253:          inx
46ea      254:
46ea c92e  255: .b:      cmp  #'.'
46ec f015  256:          beq  .strip
46ee cc6844 257:          cpy  parm_1
46f1 f010  258:          beq  .strip
46f3 995946 259:          sta  path1,y
46f6 8c5946 260:          sty  path1
46f9 c8    261:          iny
46fa a99b  262:          lda  #$9b
46fc 995946 263:          sta  path1,y
46ff f0dc  264:          beq  .a
4701 d0da  265:          bne  .a
4703      266: .strip:
4703 ad5846 267:          lda  totalpath
4706 c900  268:          cmp  #$00
4708 f00b  269:          beq  .done
470a ac5946 270:          ldy  path1
470d 88    271: .loop:   dey
470e b95946 272:          lda  path1,y
4711 c900  273:          cmp  #$00          ;'%
4713 d0f8  274:          bne  .loop
4715 8c5946 275: .done:   sty  path1
4718 a99b  276:          lda  #$9b
471a 995946 277:          sta  path1,y
471d a004  278:          ldy  #4
471f b95946 279:          lda  path1,y
4722 c900  280:          cmp  #$00          ;'%
4724 d00c  281:          bne  .exit
4726 a93e  282:          lda  #'>'
4728 995946 283:          sta  path1,y
472b ac5846 284:          ldy  totalpath
472e 88    285:          dey
472f 8c5846 286:          sty  totalpath
4732      287: .exit:
4732 60    288:          rts
4733      289: ;~~~~~
4733      290: ;next Path

```

```

4733      291: ;~~~~~
4733      292:
4733      293: next_path:
4733 ad5846 294:          lda    totalpath
4736 c900   295:          cmp    #$00
4738 d001   296:          bne    .next
473a 60    297:          rts
473b      298: .next:
473b a000   299:          ldy    #$00
473d c8    300: .a:      iny
473e b95946 301:          lda    path1,y
4741 cc5946 302:          cpy    path1
4744 f011   303:          beq    .done
4746 c900   304:          cmp    #$00          ;'%
4748 d0f3   305:          bne    .a
474a a93e   306:          lda    #'>'
474c 995946 307:          sta    path1,y
474f ac5846 308:          ldy    totalpath
4752 88     309:          dey
4753 8c5846 310:          sty    totalpath
4756 60     311:          rts
4757      312: .done:
4757 a900   313:          lda    #$00
4759 8d5846 314:          sta    totalpath
475c 60     315:          rts
475d      316:
475d      317:
475d      318: ;   Make an ASCII ($9b terminated) string out of a lth-b
475d      319: ; ~~~~~
475d      320:
475d a000   321: toascii:  ldy    #0
475f b182   322:          lda    (s_ptr),y      ;ge
4761 8d8347 323:          sta    .lth
4764 f019   324:          beq    .null
4766 c8     325:          iny
4767 ee8347 326:          inc    .lth
476a b182   327: .loop:    lda    (s_ptr),y
476c 88     328:          dey
476d 9184   329:          sta    (d_ptr),y
476f c8     330:          iny
4770 c8     331:          iny
4771 cc8347 332:          cpy    .lth
4774 d0f4   333:          bne    .loop
4776 ac8347 334: .done:    ldy    .lth
4779 88     335:          dey
477a a99b   336: .rtn:     lda    #$9b
477c 9184   337:          sta    (d_ptr),y
477e 60     338:          rts
477f a000   339: .null:    ldy    #0
4781 f0f7   340:          beq    .rtn
4783      341:
4783 01     342: .lth:     dc.b   1
4784      343:

```



```

4784      344: ;   Make a length-byte string from an ASCII ($9b termina
4784      345: ;   ~~~~~
4784      346:
4784 a000  347: tostr:      ldy  #0
4786 b182  348: .tfe:          lda  (s_ptr),y
4788 f007  349:              beq  .gotend
478a c99b  350:              cmp  #$9b
478c f003  351:              beq  .gotend
478e c8    352:              iny
478f d0f5  353:              bne  .tfe
4791 8cac47 354: .gotend:      sty  .lth      ;save l
4794 c000  355:              cpy  #0
4796 f00c  356:              beq  .rtn
4798 a000  357:              ldy  #0
479a b182  358: .loop:        lda  (s_ptr),y
479c c8    359:              iny
479d 9184  360:              sta  (d_ptr),y
479f ccac47 361:              cpy  .lth
47a2 90f6  362:              bcc  .loop
47a4 a000  363: .rtn:         ldy  #0
47a6 adac47 364:              lda  .lth
47a9 9184  365:              sta  (d_ptr),y
47ab 60    366:              rts
47ac      367:
47ac 01    368: .lth:         dc.b  1
47ad      369:
47ad      370:
47ad      371: ;   Concatenate the string at s_ptr onto the end of st
47ad      372: ;   ~~~~~
47ad      373:
47ad a000  374: concat:      ldy  #0
47af b182  375:              lda  (s_ptr),y      ;ge
47b1 8ded47 376:              sta  .src1th
47b4 f02a  377:              beq  .end      ;nu
47b6 b184  378:              lda  (d_ptr),y      ;ge
47b8 8dec47 379:              sta  .desty
47bb eeeb47 380:              inc  .srcy
47be eeec47 381:              inc  .desty
47c1 a001  382:              ldy  #1      ;al
47c3 8ceb47 383:              sty  .srcy      ; t
47c6 aceb47 384: .loop:        ldy  .srcy
47c9 b182  385:              lda  (s_ptr),y      ;ge
47cb acec47 386:              ldy  .desty
47ce 9184  387:              sta  (d_ptr),y      ;st
47d0 adeb47 388:              lda  .srcy
47d3 cded47 389:              cmp  .src1th      ;en
47d6 f008  390:              beq  .end      ;ye
47d8 eeeb47 391:              inc  .srcy
47db eeec47 392:              inc  .desty      ;up
47de d0e6  393:              bne  .loop      ;an
47e0 a000  394: .end:         ldy  #0
47e2 b184  395:              lda  (d_ptr),y      ;up
47e4 18    396:              clc              ; o

```

```

47e5 6ded47 397:          adc  .srclth
47e8 9184  398:          sta  (d_ptr),y
47ea 60    399:          rts
47eb      400:
47eb 01    401: .srcy:          dc.b  1
47ec 01    402: .desty:         dc.b  1
47ed 01    403: .srclth:        dc.b  1
47ee      404:
47ee      405:
47ee      406: ;
47ee      407: ;   Move string pointed to by s_ptr to string pointed
47ee      408: ;   ~~~~~
47ee      409:
47ee      410:
47ee a000  411: move_str:      ldy  #0
47f0 b182  412:          lda  (s_ptr),y      ;ge
47f2 a8    413:          tay              ;go
47f3 b182  414: .loop:        lda  (s_ptr),y
47f5 9184  415:          sta  (d_ptr),y
47f7 c000  416:          cpy  #0
47f9 f004  417:          beq  .done
47fb 88    418:          dey
47fc 4cf347 419:          jmp  .loop
47ff 60    420: .done:        rts
4800      421:
4800      422: ;-----
4800      423: ;prints Routeen macro
4800      424: ;this is a slow print to the screen
4800      425: ;usage jsr  prints
4800      426: ; .byte $9b," string to be printed",$ff
4800      427: ;-----
4800      428: ;
4800 68    429: prints:        pla
4801 8d1148 430:          sta  pstr+1
4804 68    431:          pla
4805 8d1248 432:          sta  pstr+2
4808 ee1148 433: prsl:         inc  pstr+1
480b d003  434:          bne  pstr
480d ee1248 435:          inc  pstr+2
4810 adffff 436: pstr:         lda  $ffff
4813 c9ff  437:          cmp  #$ff
4815 f006  438:          beq  estri
4817 20d448 439:          jsr  fast_output
481a 4c0848 440:          jmp  prsl
481d ad1248 441: estri:        lda  pstr+2
4820 48    442:          pha
4821 ad1148 443:          lda  pstr+1
4824 48    444:          pha
4825 60    445:          rts
4826      446:
4826      447: pr_byte:
4826 a200  448:          ldx  #$00
4828      449: pr_card:

```

```

4828 85d4 450:          sta  fr0
482a 86d5 451:          stx  fr0+1
482c a900 452:          lda  #$00
482e 85f2 453:          sta  cix
4830 20aad9 454:         jsr  ifp
4833 20e6d8 455:         jsr  fasc
4836 a000 456:          ldy  #$00
4838      457: .aprbloop:
4838 b1f3 458:          lda  (inbuff),y
483a 08 459:          php
483b 297f 460:          and  #$7f
483d c8 461:          iny
483e 20174a 462:         jsr  echo
4841 28 463:          plp
4842 10f4 464:          bpl  .aprbloop
4844 60 465:          rts
4845      466:
4845      467: ;-----
4845      468: ; GET KEY ROUTEEN
4845      469: ;this routeen get's one key and
4845      470: ;return it to the accumulator
4845      471: ;-----
4845      472: ;
4845 86b6 473: get_key:      stx  $b6
4847 84b7 474:          sty  $b7
4849 205148 475:         jsr  .a
484c a6b6 476:          ldx  $b6
484e a4b7 477:          ldy  $b7
4850 60 478:          rts
4851 ad25e4 479: .a:          lda  $e420+5
4854 48 480:          pha
4855 ad24e4 481:          lda  $e420+4
4858 48 482:          pha
4859 60 483:          rts
485a      484:
485a      485:
485a      486:
485a a99b 487: cr_lf:      lda  #$9b
485c 20d448 488:         jsr  fast_output
485f 60 489:          rts
4860      490:
4860      491:
4860      492: ; prints: prints string with address in PR_PTR, length by
4860      493: ; ~~~~~
4860      494:
4860      495: echos:
4860 a000 496:          ldy  #0
4862 b188 497:          lda  (pr_ptr),y      ;ge
4864 f017 498:          beq  .xpri          ;ex
4866 8d7f48 499:         sta  .maxlth
4869 c8 500:          iny
486a 8c7e48 501:         sty  .tempy
486d      502:

```

```

486d ac7e48 503: .pril:      ldy  .tempy
4870 b188 504:      lda  (pr_ptr),y
4872 20d448 505:      jsr  fast_output
4875 ee7e48 506: .prnext:    inc  .tempy
4878 ce7f48 507:      dec  .maxlth
487b d0f0 508:      bne  .pril
487d 60 509: .xpri:      rts
487e      510:
487e 01 511: .tempy:     dc.b  1
487f 01 512: .maxlth:    dc.b  1
4880      513:
4880      514:
4880      515: ;
4880      516: ; prints: prints string with address in PR_PTR, length by
4880      517: ; ~~~~~
4880      518:
4880      519: echosE:
4880 a000 520:      ldy  #0
4882 b188 521:      lda  (pr_ptr),y      ;ge
4884 f01b 522:      beq  .xpri           ;ex
4886 8da348 523:      sta  .maxlth
4889 c8 524:      iny
488a 8ca248 525:      sty  .tempy
488d      526:
488d aca248 527: .pril:      ldy  .tempy
4890 b188 528:      lda  (pr_ptr),y
4892 c900 529:      cmp  #$00
4894 f00b 530:      beq  .xpri           ; s
4896 20d448 531:      jsr  fast_output
4899 eea248 532: .prnext:    inc  .tempy
489c cea348 533:      dec  .maxlth
489f d0ec 534:      bne  .pril
48a1 60 535: .xpri:      rts
48a2      536:
48a2 01 537: .tempy:     dc.b  1
48a3 01 538: .maxlth:    dc.b  1
48a4      539:
48a4      540:
48a4      541:
48a4      542: ;
48a4      543: ;-----
48a4      544: ;actual screen handler
48a4      545: ;screen offset definitions
48a4      546: ;-----
48a4      547: scr_offset:
48a4 0000 548:      dc.w  0
48a6 2800 549:      dc.w  40
48a8 5000 550:      dc.w  80
48aa 7800 551:      dc.w 120
48ac a000 552:      dc.w 160
48ae c800 553:      dc.w 200
48b0 f000 554:      dc.w 240
48b2 1801 555:      dc.w 280

```

```

48b4 4001 556:          dc.w 320
48b6 6801 557:          dc.w 360
48b8 9001 558:          dc.w 400
48ba b801 559:          dc.w 440
48bc e001 560:          dc.w 480
48be 0802 561:          dc.w 520
48c0 3002 562:          dc.w 560
48c2 5802 563:          dc.w 600
48c4 8002 564:          dc.w 640
48c6 a802 565:          dc.w 680
48c8 d002 566:          dc.w 720
48ca f802 567:          dc.w 760
48cc 2003 568:          dc.w 800
48ce 4803 569:          dc.w 840
48d0 7003 570:          dc.w 880
48d2 9803 571:          dc.w 920
48d4      572: ;
48d4      573: fast_output:
48d4 855a 574:          sta $5a
48d6 ad0407 575:         lda sc.redirect
48d9 c902 576:          cmp #print_p
48db f01f 577:          beq .go_cio
48dd      578: .leo:
48dd adff02 579:         lda $02ff      ; CONTROL 1
48e0 c900 580:          cmp #$00
48e2 d0f9 581:          bne .leo
48e4      582:
48e4 a55a 583:          lda $5a
48e6 c920 584:          cmp #32
48e8 9004 585:          bcc .do_look
48ea c97d 586:          cmp #125
48ec 9013 587:          bcc .do_here
48ee a000 588: .do_look:      ldy #0
48f0 b9f749 589: .lookup:      lda .char_table,y
48f3 f00c 590:          beq .do_here
48f5 c55a 591:          cmp $5a
48f7 f003 592:          beq .go_cio
48f9 c8 593:          iny
48fa d0f4 594:          bne .lookup
48fc      595:
48fc      596:
48fc a55a 597: .go_cio:      lda $5a
48fe 4c074a 598:          jmp putlocal
4901      599:
4901      600:
4901 a000 601: .do_here:      ldy #0
4903 a55d 602:          lda $5d
4905 915e 603:          sta ($5e),y
4907 a55a 604:          lda $5a
4909 c99b 605:          cmp #$9b
490b f044 606:          beq .docr
490d 20d649 607: .notcr:      jsr .get_adr
4910 a55a 608:          lda $5a

```

```

4912 297f 609:      and  #$7f
4914 c920 610:      cmp  #32
4916 9007 611:      bcc  .add64
4918 c960 612:      cmp  #96
491a 9009 613:      bcc  .sub32
491c 4c2849 614:     jmp  .asis
491f 18 615: .add64:      clc
4920 6940 616:      adc  #64
4922 4c2849 617:     jmp  .asis
4925 38 618: .sub32:      sec
4926 e920 619:      sbc  #32
4928 245a 620: .asis:      bit  $5a
492a 1002 621:      bpl  .xxlate
492c 0980 622:      ora  #$80
492e a000 623: .xxlate:     ldy  #0
4930 915e 624:      sta  ($5e),y
4932 e655 625:      inc  $55
4934 e663 626:      inc  $63
4936 a555 627:      lda  $55
4938 c928 628:      cmp  #40
493a b019 629:      bcs  .next_row
493c c8 630:      iny
493d b15e 631:      lda  ($5e),y
493f 855d 632:      sta  $5d
4941 0980 633:      ora  #$80
4943 aef002 634:     ldx  752
4946 d002 635:      bne  .nocurs1
4948 915e 636:      sta  ($5e),y
494a e65e 637: .nocurs1:    inc  $5e
494c d002 638:      bne  .nooav
494e e65f 639:      inc  $5e+1
4950 60 640: .nooav:      rts
4951 a900 641: .docr:      lda  #0
4953 8563 642:      sta  $63
4955 a552 643: .next_row:   lda  $52
4957 8555 644:      sta  $55
4959 e654 645:      inc  $54
495b a454 646:      ldy  $54
495d c018 647:      cpy  #24
495f b013 648:      bcs  .scroll
4961 20d649 649:     jsr  .get_adr
4964 a000 650:      ldy  #0
4966 b15e 651:      lda  ($5e),y
4968 855d 652:      sta  $5d
496a aef002 653:     ldx  752
496d d004 654:      bne  .nocurs2
496f 0980 655:      ora  #$80
4971 915e 656:      sta  ($5e),y
4973 60 657: .nocurs2:    rts
4974 658:
4974 c654 659: .scroll:     dec  $54
4976 a558 660:      lda  $58
4978 8568 661:      sta  $68

```

```

497a 18 662:      clc
497b 6928 663:      adc  #40
497d 855e 664:      sta  $5e
497f a559 665:      lda  $58+1
4981 8569 666:      sta  $68+1
4983 6900 667:      adc  #0
4985 855f 668:      sta  $5e+1
4987 a000 669:      ldy  #0
4989 a203 670:      ldx  #3
498b b15e 671: .sloop:      lda  ($5e),y
498d 9168 672:      sta  ($68),y
498f c8 673:      iny
4990 d0f9 674:      bne  .sloop
4992 e65f 675:      inc  $5e+1
4994 e669 676:      inc  $68+1
4996 ca 677:      dex
4997 d0f2 678:      bne  .sloop
4999 b15e 679: .sloop2:      lda  ($5e),y
499b 9168 680:      sta  ($68),y
499d c8 681:      iny
499e c098 682:      cpy  #152
49a0 90f7 683:      bcc  .sloop2
49a2 a558 684:      lda  $58
49a4 18 685:      clc
49a5 6998 686:      adc  #920&$ff
49a7 855e 687:      sta  $5e
49a9 a559 688:      lda  $58+1
49ab 6903 689:      adc  #920/256
49ad 855f 690:      sta  $5e+1
49af a000 691:      ldy  #0
49b1 98 692:      tya
49b2 915e 693: .clear:      sta  ($5e),y
49b4 c8 694:      iny
49b5 c028 695:      cpy  #40
49b7 90f9 696:      bcc  .clear
49b9 aef002 697:      ldx  752
49bc d006 698:      bne  .nocurs3
49be a980 699:      lda  #$80
49c0 a455 700:      ldy  $55
49c2 915e 701:      sta  ($5e),y
49c4 a900 702: .nocurs3:      lda  #0
49c6 855d 703:      sta  $5d
49c8 a55e 704:      lda  $5e
49ca 18 705:      clc
49cb 6555 706:      adc  $55
49cd 855e 707:      sta  $5e
49cf a55f 708:      lda  $5e+1
49d1 6900 709:      adc  #0
49d3 855f 710:      sta  $5e+1
49d5 60 711:      rts
49d6 a554 712: .get_adr:      lda  $54
49d8 0a 713:      asl
49d9 a8 714:      tay

```

```

49da a558 715:      lda  $58
49dc 18 716:      clc
49dd 79a448 717:    adc  scr_offset,y
49e0 855e 718:      sta  $5e
49e2 a559 719:      lda  $58+1
49e4 79a548 720:    adc  scr_offset+1,y
49e7 855f 721:      sta  $5e+1
49e9 a55e 722:      lda  $5e
49eb 18 723:      clc
49ec 6555 724:      adc  $55
49ee 855e 725:      sta  $5e
49f0 a55f 726:      lda  $5e+1
49f2 6900 727:      adc  #0
49f4 855f 728:      sta  $5e+1
49f6 60 729:      rts
49f7 730: .char_table:
49f7 1b1c1d 731:      dc.b 27,28,29,30,31,125,126,127
49ff 9c9d9e 732:      dc.b 156,157,158,159,253,254,255
4a07 733:
4a07 734:
4a07 735: putlocal:
4a07 a20b 736:      ldx  #11
4a09 8e4203 737:      stx  $0342
4a0c a200 738:      ldx  #0
4a0e 8e4803 739:      stx  $0348
4a11 8e4903 740:      stx  $0349
4a14 4c56e4 741:      jmp  $e456
4a17 742: ;
4a17 743: ;
4a17 744:
4a17 745: ;
4a17 746: ;-----
4a17 747: ; PRINT BYTE
4a17 748: ;-----
4a17 749: echo:
4a17 85b8 750: sprint:      sta  $b8
4a19 a904 751:      lda  #$04
4a1b 8d4a03 752:      sta  $034a
4a1e 86b6 753:      stx  $b6
4a20 84b7 754:      sty  $b7
4a22 a206 755:      ldx  #$06
4a24 bd01e4 756:      lda  $e400+1,x
4a27 85b3 757:      sta  $b3
4a29 bd00e4 758:      lda  $e400,x
4a2c 85b2 759:      sta  $b2
4a2e a5b8 760:      lda  $b8
4a30 203f4a 761:      jsr  .a
4a33 a6b6 762:      ldx  $b6
4a35 a4b7 763:      ldy  $b7
4a37 a90c 764:      lda  #$0c
4a39 8d4a03 765:      sta  $034a
4a3c a5b8 766:      lda  $b8
4a3e 60 767:      rts

```



```

4a3f      768: ;
4a3f e6b2 769: .a:      inc    $b2
4a41 6cb200 770:      jmp    ($b2)
4a44 60    771:      rts
4a45      772:
4a45      773:
4a45      774: ;=====
4a45      775: ;Start of file.
4a45      776: ;
4a45      777: ;=====
4a45      778: top:
4a45 a97d 779:      lda    #$7d
4a47 20174a 780:      jsr    echo
4a4a 20a442 781:      jsr    tammy-3
4a4d      782:
4a4d 200048 783:      jsr    printsi
4a50 9b    784:      dc.b   $9b
4a51 919292 785:      dc.b   <+128>,"
4a75 9b    786:      dc.b   $9B
4a76 fca0a0 787:      dc.b   <+128>,"| Recursive Dire
4a9a 9b    788:      dc.b   $9B
4a9b fca0a0 789:      dc.b   <+128>,"| by Stephen
4abf 9b    790:      dc.b   $9B
4ac0 9a9292 791:      dc.b   <+128>,"

```