


```

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: ;      Delay.com
0000 9: ;
0000 10: ;      ~~~~~
0000 11: ;-----
0000 12: ; Notes: o This source code MAY NOT be placed for download
0000 13: ;      o "mea" is a macro that loads the address of the
0000 14: ;      into the pointer specified by the second field.
0000 15: ;-----
0000 16: ; Assembler: MADMAC (tm) ST Cross Assembler (Atari Corp)
0000 17: ;      XASM st and IBM VERSIONS
0000 18: ;-----
0000 19: ;
0000 20: ;      init design for "WASEO" sjc
0000 21: ; added no parm will delay forever
0000 22: ; added "/q" for quiet will delay and not show anything to
0000 23: ; Added pal detection to alter the tic var. 6/13/2010
0000 24: ;
0000 25: ;      Updated the screen Print Utility so it will work wi
0000 26: ; This change will also help "Print P:" 6/1
0000 27: ;
0000 28: ;
0000 29: ;
0000 30: ;
0000 31: ;
0000 32: ;
0000 33: ;=====
0000 34: com_file_name: .macro
0000 35: ;      dc.b 13,"filename.ext",$9b
0000 36: ;      dc.b 12,"Delay.com ", $9b
0000 37: ;      .endm
0000 38: ;
0000 39: ;
0000 40: ;~~~~~
0000 41: ; File revision history. Version Number in hex
0000 42: ;
0000 43: file_ver: equ $16
0000 44: ;~~~~~
0000 45: ; the Month in dec for the first time this revision
0000 46: ;was compiled
0000 47: ;
0000 48: c_month: equ 6
0000 49: ;~~~~~
0000 50: ; the day in dec for the first time this revision
0000 51: ;was compiled
0000 52: c_day: equ 19
0000 53: ;~~~~~

```

```

0000      54: ; the year in dec for the first time this revision
0000      55: ;was compiled
0000      56: c_year:          equ 2010
0000      57: ;~~~~~
0000      58: ; Type of compiler used in program
0000      59: ;
0000      60: ; 0 = unknown Compiler
0000      61: ; 1 = xasm
0000      62: ; 2 = mac_65
0000      63: ; 3 = basic
0000      64: ; 4 = compiled basic xl
0000      65: ; 5 = C65
0000      66: ; 6 = Action
0000      67: ;
0000      68: ; We can add more as time goes on!
0000      69: ;
0000      70: ;
0000      71: xasm:          equ 1
0000      72: mac_65:       equ 2
0000      73: basic:       equ 3
0000      74: basicxl:    equ 4
0000      75: c65:       equ 5
0000      76: action:      equ 6
0000      77:
0000      78: file_compiler: equ xasm
0000      79: ;~~~~~
0000      80: ; is this relocatable code ?
0000      81: ; anyother value other than 1 or 2 would be unknown
0000      82: r..yes:       equ 1
0000      83: r..no:        equ 2
0000      84: ;
0000      85: ;~~~~~
0000      86: ; Gotta define if it can be relocated
0000      87: l_relocatable: equ r..no
0000      88: ;
0000      89: ;
0000      90: ;~~~~~
0000      91: ; Gotta define if it can be relocated
0000      92: r..crl:       equ $7d
0000      93: r..crlf:      equ $9b
0000      94: r..space:     equ $20
0000      95: r..bypass:    equ $ff
0000      96: l_frstscreenbyte: equ r..bypass
0000      97: ;~~~~~
0000      98: ; The language the output file is in.
0000      99: ;
0000     100: ;
0000     101: ; 0 = Undefined
0000     102: ; 1 = English
0000     103: ; 2 = German
0000     104: ;
0000     105: r..language:   equ 1
0000     106: ;~~~~~

```

```

0000      107:
0000      108:
0000      109:          .include    equates
0000      110:          .include    globals
0000      111:          .include    macros
0000      112:
4000      113:          .org    $4000
4000      114:
4000      115: win_start:
4000      116: header_info:
4000      117:          .include    header
42a7      118:
42a7      119: start:
42a7 20f347 120:          jsr    pal_ntsc
42aa      121:
42aa a03f 122:          ldy    #LBUF
42ac b10a 123: mux_1a:          lda    (comtab),y
42ae c920 124:          cmp    #' '
42b0 f00b 125:          beq    mux_1b
42b2 c99b 126:          cmp    #$9b
42b4 f004 127:          beq    .a
42b6 c8 128:          iny
42b7 4cac42 129:          jmp    mux_1a
42ba 4c7e45 130: .a:          jmp    NO_number
42bd      131:
42bd      132: mux_1b:
42bd a200 133:          ldx    #$00
42bf c8 134:          iny
42c0      135:
42c0 b10a 136: .a:          lda    (comtab),y
42c2 20c648 137:          jsr    upcase
42c5 c948 138:          cmp    #'H'
42c7 f022 139:          beq    .help
42c9 c93f 140:          cmp    #'?'
42cb f01b 141:          beq    .help1
42cd c942 142:          cmp    #'B'
42cf f01e 143:          beq    .go_bat
42d1 c920 144:          cmp    #' '
42d3 f01d 145:          beq    .slash
42d5 9dbd46 146:          sta    mynumberstring,x
42d8 c930 147:          cmp    #'0'
42da 9024 148:          bcc    .oknum
42dc c93a 149:          cmp    #'9' +1
42de b020 150:          bcs    .oknum
42e0 e8 151:          inx
42e1 8eee42 152:          stx    .holdx
42e4 c8 153:          iny
42e5 4cc042 154:          jmp    .a
42e8      155:
42e8 4c7243 156: .help1:          jmp    mux_man1
42eb 4c6e43 157: .help:          jmp    mux_man3
42ee 00 158: .holdx:          dc.b    0
42ef 4c6b43 159: .go_bat:          jmp    go_bat

```

```

42f2      160:
42f2      161: .slash:
42f2 c8    162:          iny
42f3 c8    163:          iny
42f4 b10a   164:          lda  (comtab),y
42f6 20c648 165:          jsr  upcase
42f9 8d7143 166:          sta  scr_hold
42fc c951   167:          cmp  #'Q'
42fe f003   168:          beq  .oknum1
4300      169: .oknum:
4300 20a442 170:          jsr  start-3
4303      171:
4303 ae42   172: .oknum1:      lda  .holdx
4306 e8     173:          inx
4307 a99b   174:          lda  #$9b
4309 9dbd46 175:          sta  mynumberstring,x
430c      176:
430c 209446 177:          jsr  get_number_cli
430f 8dbc47 178:          sta  howlongtohold
4312 c900   179:          cmp  #$00
4314 d003   180:          bne  mux_1c
4316 4c7e45 181:          jmp  NO_number
4319      182:
4319      183: mux_1c:
4319 a9ff   184:          lda  #$ff
431b 8dfc02 185:          sta  764
431e 201a45 186:          jsr  setup
4321      187: .a:
4321 a906   188:          lda  #consol_start
4323 cd1fd0 189:          cmp  $d01f
4326 f04a   190:          beq  mux_man1
4328 a903   191:          lda  #consol_option
432a cd1fd0 192:          cmp  $d01f
432d f03f   193:          beq  mux_man3
432f      194:
432f adfc02 195:          lda  764
4332 c93f   196:          cmp  #$3f      ;'A'
4334 f019   197:          beq  .abort
4336 c97f   198:          cmp  #$7f
4338 f015   199:          beq  .abort
433a c9ff   200:          cmp  #$ff
433c d014   201:          bne  .out
433e adbc47 202:          lda  howlongtohold
4341 cdbd47 203:          cmp  reference
4344 f00c   204:          beq  .out
4346 20be47 205:          jsr  delay
4349 cebc47 206:          dec  howlongtohold
434c 4c2143 207:          jmp  .a
434f      208:
434f 4c74e4 209: .abort:          jmp  $e474
4352      210:
4352      211: .out:
4352      212:

```

```
4352 a9ff 213:          lda  #$ff
4354 8dfc02 214:          sta  764
4357 a900 215:          lda  #$00
4359 8df002 216:          sta  752
435c      217:
435c ad7143 218:          lda  scr_hold
435f c951 219:          cmp  #'Q'
4361 f007 220:          beq  .exit
4363      221:
4363 207948 222:          jsr  printsi
4366 9b9b9b 223:          dc.b  $9b,$9b,$9b,-1
436a 60    224: .exit:      rts
436b      225:
436b      226:
436b      227:
436b 4cc645 228: go_bat:      jmp  mux_man2
436e      229:
436e 4cf248 230: mux_man3:    jmp  my_file_notice
4371      231:
4371 00     232: scr_hold:    dc.b  0
4372      233:
4372 204a48 234: mux_man1:    jsr  bell
4375 207948 235:          jsr  printsi
4378 7d9b9b 236:          dc.b  $7d,$9b,$9b
437b 20     237:          dc.b  " "
437c 919292 238:          dc.b  <+128>,"
439e 9b     239:          dc.b  $9B
439f 20     240:          dc.b  " "
43a0 fca0c4 241:          dc.b  <+128>,"| Delay.com  Batch
43c2 9b     242:          dc.b  $9B
43c3 20     243:          dc.b  " "
43c4 fca0d7 244:          dc.b  <+128>,"| Written by: Steph
43e6 9b     245:          dc.b  $9B
43e7 20     246:          dc.b  " "
43e8 9a9292 247:          dc.b  <+128>,"
```