

```

;
;
;*****
;*
;*          This program put new cp/m
;*          in disk or in hard disk
;*
;*****
;
;          title PutSys 1.61 for NE CP/M with Hard-Disk
;          subttl Copyright Studio Lg, Genova - Last rev 15/08/1984 10:40
;          Programmers: Martino Stefano & Gallerani Paolo
;
0011      vers      equ      17
;
;
0001      true      equ      1
0000      false     equ      0
;
0000      copyrig   equ      false      ; flag for copyright message
;
;          Include ASCII.LIB      ; ascii equivalents
;
;
;*****
;*
;*          ASCII EQUIVALENTS
;*
;*****
;
0007      C      bell      equ      'G'- '@'      ; ring beeper
0008      C      backsp    equ      'H'- '@'      ; back space char.
0009      C      tab       equ      'I'- '@'      ; tabulation char.
000A      C      lf        equ      'J'- '@'      ; line-feed char.
000C      C      ffeed     equ      'L'- '@'      ; form feed char.
000D      C      cr        equ      'M'- '@'      ; carriage-return char.
0013      C      pfx       equ      'S'- '@'      ; attributes pfx
0048      C      rever     equ      'H'           ; Reverse On (^SH)
0043      C      flash     equ      'C'           ; Flash On (^SC)
0040      C      norm      equ      '@'           ; Normal (^SE)
0020      C      space     equ      ' '           ; space char.
0024      C      endmsg    equ      '$'           ; end of print message
;
;
;          page      62

```

```

C      ;          Include ROMENTRY.LIB      ; Rom entry points
C      ;
C      ;*****
C      ;*                                     *
C      ;*           Rom routines address     *
C      ;*                                     *
C      ;*****
C      ;
F000   C      rom    equ    0F000h          ; (--- rom starting address
F003   C      cin    equ    rom+3          ; console input
F006   C      cout   equ    rom+6          ; console output
F009   C      csts   equ    rom+9          ; console status
F00C   C      lout   equ    rom+12         ; printer output
F00F   C      lsts   equ    rom+15         ; printer status
F012   C      fdias   equ    rom+18        ; fdd I/O 128 byte
F015   C      fdiod   equ    rom+21        ; fdd I/O 256 byte
F018   C      wdini   equ    rom+24        ; wdd initialization
F01B   C      wdio    equ    rom+27        ; wdd I/O 256 byte
F01E   C      strout  equ    rom+30        ; print string .DE until $
F01E   C      print  equ    strout         ; sinonime
F021   C      bootrom equ    rom+33        ; load BIOS and go to wbootc
F024   C      printat equ    rom+36        ; print str. -> DE at -> HL cursor
F027   C      movcurs equ    rom+39        ; move cursor at -> HL
F02A   C      vidinit equ    rom+42        ; initialize video
F02D   C      CompFlg equ    rom+45        ; Version Number
C      ;
C      ;
C      ;*****
C      ;*           Hard Disk Status Port     *
C      ;*                                     *
C      ;*****
C      ;
00B9   C      rport1 equ    0b9h           ; sasi read port 1 - read status
C      ;
C      ;*****
C      ;*                                     *
C      ;*           BDOS FUNCTIONS             *
C      ;*                                     *
C      ;*****
C      ;
0005   C      bdos    equ    0005          ; bdos entry point
C      ;
0001   C      bcinp   equ    01            ; input one char. from console
0002   C      bcout   equ    02            ; output one char. to console
0009   C      bprint  equ    09            ; output buffer to console
000A   C      bstrinp  equ    10           ; input buffer from console
000F   C      bopen   equ    15            ; open file
0010   C      bclose  equ    16            ; close file
0014   C      bread   equ    20            ; read next sector
001A   C      dma     equ    26            ; set dma address
C      ;
C      ;
page

```

0000'

0100 31 02AC
 0103 C3 02F7

```

;
;
;
*****
* P U T S Y S H *
*      Set stack and go to setup      *
*                                     *
*****
;
;
Aseq
Org 100h ; On TPA
;
ld sp,stack ; set stack pointer
jp putsys ; go to put system
;
;
;
*****
* Message Area *
*****
;
;
if copyrig
defb ' COPYRIGHT (c) 1983, 1984 by STUDIO Lg, Genova, ITALY '
endif
;
;
;

```

0106 0D 0A 07 49
 010A 6E 63 6F 6D
 010E 70 61 74 69
 0112 62 6C 65 20
 0116 56 65 72 73
 011A 69 6F 6E 20
 011E 4F 66 20 45
 0122 50 52 4F 4D
 0126 24

No4.2: defb cr,lf,bell,'Incompatible Version Of EPROM',endmsg

0127
 0127 0D 0A 55 6E
 012B 72 65 63 6F
 012F 76 65 72 61
 0133 62 6C 65 20
 0137 65 72 72 6F
 013B 72 20 69 6E
 013F 20 68 61 72
 0143 64 20 64 69
 0147 73 6B 0D 0A
 014B
 014B 0A 53 65 74
 014F 20 6E 65 77
 0153 20 73 79 73
 0157 74 65 6D 20
 015B 64 69 73 6B
 015F 65 74 74 65
 0163 20 69 6E 20
 0167 64 72 69 76
 016B 65 20 41 2C
 016F 20

```

;
;
;
wdfatmsg:
defb cr,lf,'Unrecoverable error in hard disk',cr,lf
;
;
;
setnew:
defb lf,'Set new system diskette in drive A, '
;
;
;

```

```

0170 74 68 65 6E      defb  'then push return.',endmsg
0174 20 70 75 73
0178 68 20 72 65
017C 74 75 72 6E
0180 2E 24

;
;
;
;
0182      inimg:
0182 0D 0A 13 48      defb  cr,lf,pfx,'HPUTSYS Vers. ',vers/10+'0','.',vers mod 10+'0',pfx,'@',cr,lf
0186 50 55 54 53
018A 59 53 20 56
018E 65 72 73 2E
0192 20 31 2E 37
0196 13 40 0D 0A
019A 0D 0A 50 75      defb  cr,lf,'Put system to Hard disk or to Floppy disk (H/F) ?',endmsg
019E 74 20 73 79
01A2 73 74 65 6D
01A6 20 74 6F 20
01AA 48 61 72 64
01AE 20 64 69 73
01B2 6B 20 6F 72
01B6 20 74 6F 20
01BA 46 6C 6F 70
01BE 70 79 20 64
01C2 69 73 6B 20
01C6 28 48 2F 46
01CA 29 20 3F 24

;
;
;
;
01CE      sizemsg:
01CE 0D 0A 0A 43      defb  cr,lf,lf,'Choose your system configuration size,'
01D2 68 6F 6F 73
01D6 65 20 79 6F
01DA 75 72 20 73
01DE 79 73 74 65
01E2 6D 20 63 6F
01E6 6E 66 69 67
01EA 75 72 61 74
01EE 69 6F 6E 20
01F2 73 69 7A 65
01F6 2C
01F7 0D 0A 66 72      defb  cr,lf,'from following ones: 32,40,48,56,60 kbyte ',endmsg
01FB 6F 6D 20 66
01FF 6F 6C 6C 6F
0203 77 69 6E 67
0207 20 6F 6E 65
020B 73 3A 20 33
020F 32 2C 34 30
0213 2C 34 38 2C
0217 35 36 2C 36
021B 30 20 6B 62
021F 79 74 65 20
0223 24

;
;
;
;
0224      fdrdymsg:
0224 0D 0A 50 75      defb  cr,lf,'Push any key when disk has been set in drive A [^C to exit] ',endmsg

```

0228 73 68 20 61
022C 6E 79 20 6B
0230 65 79 20 77
0234 68 65 6E 20
0238 64 69 73 6B
023C 20 68 61 73
0240 20 62 65 65
0244 6E 20 73 65
0248 74 20 69 6E
024C 20 64 72 69
0250 76 65 20 41
0254 20 5B 5E 43
0258 20 74 6F 20
025C 65 78 69 74
0260 5D 20 24

0263 0D 0A 07 46
0267 69 6C 65 20
026B 6E 6F 74 20
026F 66 6F 75 6E
0273 64 20 3A 20
0277 24

027B 0D 0A 07 44
027C 69 73 6B 20
0280 49 2F 4F 20
0284 65 72 72 6F
0288 72 0D 0A 24

```
;
nofilmsg:
    defb    cr,lf,bell,'File not found : ',endmsg
```

```
;
errmsg:
    defb    cr,lf,bell,'Disk I/O error',cr,lf,endmsg
```

```
;
page
```



```

;
;
;
;*****
;*          Data Area          *
;*****
;
;      defs      32          ; stack data areas
02BC      stack   equ      $          ; stack top
02AC
;
;
;      putflag:
02AC      defs      1          ; put system flag
02AC
;
;      cpmadat:
02AD      defs      2          ;
02AD
;
;
;      conbuff:
02AF      02      defb      2          ; two char. for input
02AF      defs      8          ; free areas
02B0
;
;
;      fcb:
02B8      00      defb      0          ; logical disk
02B8      53 59 53 33      defb      'SYS32  DVR' ; name of file to open
02B9      32 20 20 20
02BD      4F 56 52
02C1
02C4      defs      32          ; file control block areas
;
;
;
;
;      xlt:
02E4      ; sector translate table
02E4      01 07 03 09      defb      1,7,3,9,5,2,8,4,10,6
02E8      05 02 08 04
02EC      0A 06
02EE      FF      defb      255          ; flag for end track
;
; Disk I/O Table
;
;      PutTab:
02EF      00      Po4 000: defb      0          ; unit 0 - side 0
02EF      0000      PutTrk: defw      0          ; track 0000
02F0      01      PutSec: defb      1          ; sector 0
02F2      0446      PutDma: defw      cpmbuff          ; dma adrs
02F3      01      defb      1          ; write code
02F5      1D      defb      29          ; block count (29 sector 256 byte)
02F6      ;          ; for wdd putsys
;
;      page

```

```

02F7
02F7 3A F02D
02FA FE 2A
02FC 38 5C
02FE DB B9
0300 E6 E0
0302 20 20

0304 11 0182
0307 CD 041C
030A
030A CD 040B
030D FE 20
030F 38 F9
0311 CB AF
0313 FE 46
0315 28 0D
0317 FE 48
0319 3E 00
031B 28 09

031D
031D 1E 08
031F CD 0417
0322 18 E6

0324
0324 3E 01
0326
0326 32 02AC
0329
0329 11 01CE
032C CD 041C
032F CD 0421
0332 11 02B0
0335 1A
0336 FE 02
0338 20 EF
033A 13
033B 1A
033C 32 02BC
033F 13
0340 1A
0341 32 02BD
0344 CD 042E
0347 3C
0348 20 1C

034A
034A 11 0263

;
;
;*****
;* PUTSYS *
;*****
;
putsys:
    ld    a,(CompFlg)    ; Check for proper version
    cp    42             ; must be greater 42
    jr    c,BadVer       ; else error
    in    a,(rport1)     ; load cntlr status bits
    and    11100000b     ; mask bit 5,6,7. Hard disk exist ?
    jr    nz,puts01      ; no, then put system to floppy disk

;
    ld    de,inimsg      ; D.E = initial message
    call  printst        ; print it

puts0:
    call  cinput         ; wait one char.
    cp    ' '            ; ( ' ' ?
    jr    c,puts0       ; yes, ignore
    res    5,a           ; convert up-case
    cp    'F'           ; floppy disk ?
    jr    z,puts01      ; yes, then go to fdd system put
    cp    'H'           ; hard disk ?
    ld    a,0           ; set put to wdd
    jr    z,puts2       ; continue
;

invalid:
    ld    e,backsp      ; back space char.
    call  coutput       ; printst it
    jr    puts0         ;

;
puts01:
    ld    a,1           ; set put to floppy

puts2:
    ld    (putflag),a    ; set put system flag (0 = hard, 1 = floppy)

puts2.1:
    ld    de,sizemsg     ; D.E = size message
    call  printst        ; print it
    call  strinput       ; wait two char.
    ld    de,conbuff+1   ; point to # char
    ld    a,(de)         ; get #
    cp    2             ; 2 char
    jr    nz,puts2.1     ; no, ignore
    inc    de           ; point to char
    ld    a,(de)         ; get 1' char.
    ld    (fcb+4),a      ; put to fcb
    inc    de           ; point to 2'
    ld    a,(de)         ; get 2' char.
    ld    (fcb+5),a      ; put to fcb
    call  open           ; open SYBxx.OVR
    inc    a             ; successful ?
    jr    nz,puts3      ; yes, then count
;

nofile:
    ld    de,nofilmsg    ; D.E = no file message

```

```
0340 CD 041C      call printst      ; print it
0350 3E 24        ld a,'$'          ; $ = string terminator
0352 32 02C4      ld (fcb+12),a      ; put it at end of fcb
0355 11 02B9      ld de,fcb+1       ; point top char. of fcb
0358 18 03        jr err            ; print & return to cp/m
;
035A BadVer:
035A 11 0106      ld de,No4.2        ; No 4.2 Eprom
035D err:
035D CD 041C      call printst      ; print fcb+1
0360 CD 0410      call crlf         ; print cr & lf
0363 C3 0000      jp 0               ; return to cp/m
;
page
```



```

0366                ;
0366      06 3A      puts3:  ld      b,2+44+12      ; IPL+(ccp+bdos)+bios sector count
0368      11 0446    ld      de,cpmbuff          ; load cp/m buffer
0368                puts4:
0368      ED 53 02AD  ld      (cpmdat),de        ;
036F      C5        push    bc                  ; save counter
0370      CD 0429    call    SetDma              ; set new dma
0373      CD 043E    call    fread               ; read next sector
0376      B7        or      a                   ; successful ?
0377      20 11      jr      nz,rderr            ; no, then error
0379      2A 02AD    ld      hl,(cpmdat)         ; get old dma address
037C      11 0080    ld      de,128             ; cp/m sector size
037F      19        add     hl,de                ; new dma
0380      EB        ex      de,hl                ; de = next dma addr
0381      C1        pop     bc                  ; restore counter
0382      10 E7      djnz   puts4                ; loop for all
                ;
0384                puts5:
0384      CD 0436    call    close                ; close file SYSxx.OVR
0387      3C        inc     a                   ; successful ?
0388      20 16      jr      nz,puts6            ; yes, then count
                ;
038A                rderr:
038A      11 0278    ld      de,errmsg          ; print error
038D      CD 041C    call    printst             ; message and
0390      11 014B    ld      de,setnew          ; set new message
0393      CD 041C    PrMsg: call    printst      ; print it
0396                waitcr:
0396      CD 040B    call    cinput              ; wait a key
0399      FE 0D      cp      cr                 ; cr ?
039B      20 F9      jr      nz,waitcr          ; no, ignore
039D      C3 F021    jp      bootrom           ; A()0, boot from floppy
                ;
03A0                puts6:
                ; at this point, IPL + cp/m has been loaded in cp/m buffer
                ;
03A0      3A 02AC    ld      a,(putflag)        ; load put system flag
03A3      B7        or      a                   ; test for hard or for floppy
03A4      20 12      jr      nz,fddput         ; no zero, then go to fdd put system
                ;
                page
    
```

03A6

03A6 CD F018
 03A9 21 02EF
 03AC CD F01B
 03AF 11 0127
 03B2 B7
 03B3 20 DE
 03B5 C3 03FF

```

;
;
;*****
;* W D D P U T                                     *
;*          Put IPL + CP/M in Hard Disk 0          *
;*****
;
wddput:
; put new cp/m in hard disk
call wdini          ; initialize wdd
ld hl,PutTab        ; H.L = wdd put system table
call wdio           ; write cp/m
ld de,wdfatmsg       ; Hard Msg Pointer
or a                ; error occurs ?
jr nz,PrMsg          ; yes, then go to error
jp EndPut            ; done, go to boot
;
page
    
```

```

;
;
;*****
;* F D D P U T *
;* Put IPL + CP/M in Floppy Disk 0 *
;*****
;
fddput:
03B8      ld      de,fdrdymsg      ; DE = fdd ready message
03B8      11 0224      call      printst      ; print it
03BB      CD 041C      call      cinput      ; wait one char.
03BE      CD 040B      cp      'C'-'@'      ; is cntrl C ?
03C1      FE 03      jr      z,endfddput      ; yes, then exit
03C3      2B 3A

;
03C5      AF      xor      a      ; clear accumulator
03C6      32 02F0      ld      (PutTrk),a      ; set track 0
03C9      21 0446      ld      hl,cpmbuff      ; HL = cp/m buffer top adrs
03CC      22 02F3      ld      (PutDma),hl      ; set dma to cp/m buff top adrs

;
fddpu0:
03CF      11 02E4      ld      de,xlt      ; DE = sector translate table
03D2      fddpul:
03D2      1A      ld      a,(de)      ; load physical sector
03D3      FE FF      cp      255      ; end track ?
03D5      2B 1B      jr      z,nextrk      ; yes, then point to next track
03D7      32 02F2      ld      (PutSec),a      ; set sector number
03DA      21 02EF      ld      hl,PutTab      ; HL = fdd put system table
03DD      D5      push     de      ; save xlt pointer
03DE      CD F015      call     fdiod      ; write/one sector
03E1      B7      or      a      ; error occurs ?
03E2      D1      pop      de      ; restore xlt pointer
03E3      20 A5      jr      nz,rderr      ; yes, then go to error
03E5      2A 02F3      ld      hl,(PutDma)      ; load currently dma adrs
03E8      01 0100      ld      bc,256      ; BC = sector size
03EB      09      add      hl,bc      ; point to next dma adrs
03EC      22 02F3      ld      (PutDma),hl      ; set next dma adrs
03EF      13      inc      de      ; inc. xlt pointer
03F0      1B E0      jr      fddpul      ; count for next sector
03F2      nextrk:
03F2      3A 02F0      ld      a,(PutTrk)      ; load currently track number
03F5      3C      inc      a      ; point to next track
03F6      32 02F0      ld      (PutTrk),a      ; set next track number
03F9      FE 03      cp      3      ; all three track are written ?
03FB      20 D2      jr      nz,fddpu0      ; no, then write next track
03FD      1B B9      jr      fddput      ; and count for another disk

;
;
03FF      endfddput:
03FF      EndPut:
; new cp/m has been written on Hard or floppy disk
;
03FF      DB B9      in      a,(rport1)      ; load cntlr status bits
0401      E6 E0      and      11100000b      ; mask bit 5,6,7. Hard disk exist ?
0403      CA F021      jp      z,bootrom      ; yes, then reload system to hard disk
0406      3E 01      ld      a,1      ; flag for load cp/m to floppy
0408      C3 F021      jp      bootrom      ; jump to boot rom
    
```

;
page

```

;
;
;*****
;*          BDOS FUNCTIONS          *
;*****
;
040B      cinput: ; Console input in A reg
;
040B      OE 01      ld      c,bcinp
040D      C3 0005      jp      bdos
;
0410      crlf:
;
;      ; Print Cr, Lf
;
0410      1E 0D      ld      e,cr
0412      CD 0417      call   coutput
0415      1E 0A      ld      e,lf
;
0417      coutput:
;
;      ; Print Char in A reg
;
0417      OE 02      ld      c,bcout
0419      C3 0005      jp      bdos
;
041C      printst: ; Print string pointed by DE
;
041C      OE 09      ld      c,bprint
041E      C3 0005      jp      bdos
;
0421      strinput: ; Read input from console (DE.)
;
0421      11 02AF      ld      de,conbuff
0424      OE 0A      ld      c,bstrinp
0426      C3 0005      jp      bdos
;
;      page
    
```



```

;
; *** Disk Functions ***
;
0429      SetDma: ; Set DMA add from DE
;
0429      0E 1A      ld      c,dma
042B      C3 0005      jp      bdos
;
042E      open: ; Open File (DE.FCB)
;
042E      11 02B8      ld      de,fcf
0431      0E 0F      ld      c,bopen
0433      C3 0005      jp      bdos
;
0436      close: ; Close File (DE.FCB)
;
0436      11 02B8      ld      de,fcf
0439      0E 10      ld      c,bclose
043B      C3 0005      jp      bdos
;
043E      fread: ; Read Seq (DE.FCB)
;
043E      11 02B8      ld      de,fcf
0441      0E 14      ld      c,bread
0443      C3 0005      jp      bdos
;
;
0446      cpmbuff equ  $          ; space for IPL+(ccp+bdos)+bios
;
;
;      end      100h          ; end of this program
```

Macros:

Symbols:

BACKSP	0008	BADVER	035A	BCINP	0001	BCLOSE	0010
BCOUT	0002	BDOS	0005	BELL	0007	BOOTRD	F021
BOPEN	000F	BPRINT	0009	BREAD	0014	BSTRIN	000A
CIN	F003	CINPUL	040B	CLOSE	0436	COMPFL	F02D
CONBUF	02AF	COPYRI	0000	COUT	F006	COUTPU	0417
CPMBUF	0446	CPMDAT	02AD	CR	000D	CRLF	0410
CSTS	F009	DMA	001A	ENDFDD	03FF	ENDMSG	0024
ENDPUT	03FF	ERR	035D	ERRMSG	0278	FALSE	0000
FCB	02B8	FDDPU0	03CF	FDDPU1	03D2	FDDPUT	03B8
FDIOD	F015	FDIOS	F012	FDRDYM	0224	FFED	000C
FLASH	0043	FREAD	043E	INMSG	0182	INVALI	031D
LF	000A	LOUT	F00C	LSTS	F00F	MDVCUR	F027
NEXTRK	03F2	ND4.2	0106	NOFILE	034A	NOFILM	0263
NORM	0040	OPEN	042E	PFX	0013	PRINT	F01E
PRINTA	F024	PRINTS	041C	PRMSG	0393	PUTDMA	02F3
PUTFLA	02AC	PUTS0	030A	PUTS01	0324	PUTS2	0326
PUTS2.	0329	PUTS3	0366	PUTS4	036B	PUTS50	0384
PUTS6	03A0	PUTSEC	02F2	PUTSYS	02F7	PUTTAB	02EF
PUTTRK	02F0	RDERR	03BA	REVER	0048	ROM	F000
RPORT1	00B9	SETDMA	0429	SETNEW	0148	SIZEMS	01CE
SPACE	0020	STACK	02AC	STRINP	0421	STROUT	F01E
TAB	0009	TRUE	0001	VERS	0011	VIDINI	F02A
WAITCR	0396	WDDPUT	03A6	WDFATM	0127	WDINI	F018
WDIO	F01B	XLT	02E4				

No Fatal error(s)