

# CHR Tape to Disk Transfer 1.0

```
;-----  
;CHR Tape to Disk Transfer 1.0  
;  
;(C)2005 Nostalgia  
;  
;By Fungus / Nostalgia  
;  
;Worked on Firelord, Space Trooper  
;-----  
  
    *= $0801  
  
    ;2000 sys2061  
  
    .byte $0b,$08,$d0,$07,$9e  
    .byte $32,$30,$36,$31,$00  
    .byte $00,$00  
  
    jsr $e453    ;reset basic pnt  
  
    lda $ba  
    sta temp    ;save drv #  
    lda #$0b  
    sta $d011  
    lda #$80  
    sta $0291  
    lda #$17  
    sta $d018  
    lda #$00  
    sta $9d  
    sta $d020  
    sta $d021  
    sta $c0  
    lda #$05  
    sta $0286  
    lda #$2f  
    sta $00  
    lda #$37  
    sta $01  
    lda #<msg1  
    ldx #>msg1  
    jsr print  
    lda #$1b  
    sta $d011  
    lda #$ef  
    cmp $dc01
```

```
    bne *-3
    lda #00
    sta $0289

    jsr lod      ;get header
    jsr namer    ;get the name
    jsr led      ;calc length and
                ;load file

    lda $01
    ora #20
    sta $01

xl   ldx #00      ;cracker init
    lda loader,x
    sta $0630,x
    lda loader+$0100,x
    sta $0730,x
    lda #00
    sta $da30,x
    sta $db00,x
    inx
    bne xl

cl   ldx #27      ;clr status
    lda #20
    sta $0608,x
    dex
    bpl cl

nl   ldx #0f      ;save name
    lda name,x
    sta $0340,x
    dex
    bpl nl
    jmp $0630

lod  lda #01      ;load header
    tax
    tay
    jsr $ffb8
    lda #00
    jsr $ffbd
    lda #00
    sta $c0
    sta $93
    jsr $f7d7
    jmp $f84f

led  lda #00
    sta $c3
    lda #18
```

```
        sta $c4

        lda $033f      ;file length calc
        sbc $033d
        tax
        lda $0340
        sbc $033e
        tay
        clc
        txa
        adc $c3        ;end address calc
        sta $ae
        tya
        adc $c4
        sta $af
        lda $c3
        sta $c1
        lda $c4
        sta $c2
ex      jmp $f84f

namer   ldy #$00      ;name file
        ldx #$00
dname   lda $0341,y
        bmi skip
        beq skip
        sta name,x
        sta $0610,x
        inx
skip     iny
        cpy #$10
        bne dname

        lda $0353      ;get parameter
        sta parm
        rts

print   sta $fb      ;print routine
        stx $fc
        ldy #$00
ploop   lda ($fb),y
        beq dprint
        jsr $ffd2
        inc $fb
        bne ploop
        inc $fc
        bne ploop
dprint  rts

parm    .byte $00
```



```

        lda #$24
        sta $01
        lda #$00
        ldx #$00
        ldy #$f8
l1      sta $0800,x
        inx
        bne l1
        inc l1+2
        dey
        bne l1
        lda #$27
        sta $01

start   lda #<rs
        sta $0328    ;r/s vector
        lda #>rs
        sta $0329

res     cli

        ldy #$00
        sty $c6      ;key buff
        sty $c0      ;motor on
        sty $02      ;eor val
        lda $d011    ;vic off
        and #$ef
        sta $d011

i12ca  dex          ;wait!
        bne i12ca
        dey
        bne i12ca
        sei
        jmp i1351    ;load

i12d4  lda $dc0d     ;get bit
        and #$10
        beq i12d4
        lda $dd0d
        stx $dd07
        lsr a
        lsr a
        lda #$19
        sta $dd0f
        rts

i1351  sei          ;load
mod    lda #$fa
        sta $dd06    ;timer low
        ldx #$01

i1359  jsr i12d4    ;get bit
        rol $f7      ;00 to start

```

```
    lda $f7
    cmp #$63
    bne i1359      ;sync
    ldy #$64
i1366  jsr i13e7      ;get byte
    cmp #$63
i136d  beq i1366      ;sync
    cpy $f7       ;sync
    bne i1359
    jsr i13e7      ;get byte
    iny
    bne i136d      ;skip $9d bytes
    cmp #$00
    beq i1351      ;try again
i137b  jsr i13e7      ;get byte
    sta !$2b,y     ;002b-0034
    sta !$f9,y     ;00f9-0102
    iny
    cpy #$0a
    bne i137b
    ldy #$00
    sty $90        ;st
    sty $02        ;eor val
i138f  jsr i13e7      ;get byte
    sta ($f9),y
    eor $02
    sta $02
    inc $f9
    bne i139e
i139e  inc $fa
    lda $f9
    cmp $2d
    lda $fa
    sbc $2e
    bcc i138f      ;load loop
    jsr i13e7      ;get byte
    iny
    sty $c0        ;motor off
    cli
    clc
    lda #$00
    sta $02a0      ;irqtmp! why?
    jsr $fc93      ;reset irq
    jsr $e453      ;reset basic pnts

mod1   jsr sa
    jsr se
mod2   jmp res
    jmp ok

i13e7  lda #$07      ;get byte
```

```
i13eb    sta $f8
        jsr i12d4
        rol $f7
        inc $d020
        dec $f8
        bpl i13eb
        lda $f7
        rts

sa       lda $2b
        sta $03fc
        lda $2c
        sta $03fd
        lda #$2c
        sta mod1
        rts

se       lda $31
        bne more
        lda $2d
        sta $fb
        lda $2e
        sta $fc
        lda $03fc
        sta $ac
        lda $03fd
        sta $ad
        lda $2f
        sta $03fc    ;jmp addy
        lda $30
        sta $03fd
        lda #$2c
        sta mod2

more     rts

rs       lda #$80
        ora $91
        jmp $f6ef    ;bypass run/stop

ok       lda #$00    ;saver
        sta $d020
        ldx #$07

dl       lda st,x    ;save message
        sta $0608,x
        dex
        bpl dl

        lda #$01    ;open file for
        ldx drv     ;saving
        ldy #$01
        jsr $ffba
```

```
        lda #$10
        ldx #$40
        ldy #$03
        jsr $ffbd
        jsr $ffc0
        ldx #$01
        jsr $ffc9
        lda $ac
        jsr $ffa8
        lda $ad
        jsr $ffa8
        ldy #$00
saveb   sei           ;save loop
        inc $01
        lda ($ac),y
        dec $01
        jsr $ffa8
        inc $d020
        dec $d020
        inc $ac
        bne b1
        inc $ad
b1      lda $ac
        cmp $fb
        lda $ad
        sbc $fc
        bcc saveb
        lda #$01     ;close and reset
        jsr $ffc3
        sei
        lda #$2f
        sta $00
        lda #$37
        sta $01
        inc $d020
        jmp *-3

st      .text "saving "
drv     .byte $00
```

From:  
<https://codebase64.org/> - **Codebase 64 wiki**

Permanent link:  
[https://codebase64.org/doku.php?id=base:chr\\_tape\\_to\\_disk\\_transfer\\_1.0](https://codebase64.org/doku.php?id=base:chr_tape_to_disk_transfer_1.0)

Last update: **2015-04-17 04:30**

