

# ROM-Load

```
;-----  
;Tape Xfer / Remember  
;  
;By Fungus / Remember  
;  
;(c) 1999 Sporesoft LTD.  
;  
;Works on unprotected tapes for  
;Any Commodore System  
;-----  
  
    *= $0801  
  
    .byte $0b,$08,$cf,$07,$9e  
    .byte $32,$30,$36,$31,$00  
    .byte $00,$00  
  
    lda #$0b  
    sta $d011  
    lda #$c1  
    sta $0318  
    lda #$80  
    sta $0291  
    lda #$17  
    sta $d018  
    lda #$00  
    sta $9d  
    sta $d020  
    sta $d021  
    sta $c0  
    lda #$00  
    sta $0286  
    lda #$2f  
    sta $00  
    lda #$27  
    sta $01  
    lda #<msg1  
    ldx #>msg1  
    jsr print  
    lda #$1b  
    sta $d011  
    lda #$00  
    sta $0289  
    tax  
ll   lda loader,x  
     sta $0590,x  
     lda loader+$0100,x
```

```

    sta $0690,x
    inx
    bne ll
    jmp $0590
msg1
    .byte $90,$93,$1e
    .text "AAAAAAAAAAAAAAAAAAAA"
    .text "AAAAAAAAAAAAAAAAAAAA"
    .text "Header and File Tran"
    .text "sfer 1.0 / Remember"
    .text "AAAAAAAAAAAAAAAAAAAA"
    .text "AAAAAAAAAAAAAAAAAAAA"
    .text "
           Completely"
    .text " Automatic      "
    .text "           Put a fresh di"
    .text "sk in drive 8.      "
    .text "           Rewind tape, a"
    .text "nd press PLAY.      "
    .text "
           "
    .text "
           "
    .text "   Press SPACE to be"
    .text "gin transferring.    "
    .text "AAAAAAAAAAAAAAAAAAAA"
    .text "AAAAAAAAAAAAAAAAAAAA"
    .text "
           "
    .text "
           "
    .byte $00

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

loader = *
       *= $0590
       .offs loader-*

main   lda #$ef
       cmp $dc01
       bne *-3
       jsr ton
       jsr dly
       jsr lod
       jsr tof

```

```
      jsr namer
      ldx #$0f
n1l   lda name1,x
      sta $0570,x
      dex
      bpl n1l
      ldx #$03
zp1   lda $c1,x
      sta buffer,x
      lda $ac,x
      sta buffer+$04,x
      dex
      bpl zp1
      ldx #<name1
      ldy #>name1
      jsr sav
zp2   lda buffer,x
      sta $c1,x
      lda buffer+$04,x
      sta $ac,x
      dex
      bpl zp2
      jsr ton
      jsr led
      jsr tof
n2l   ldx #$0f
      lda name2,x
      sta $0570,x
      dex
      bpl n2l
      ldx #<name2
      ldy #>name2
      jsr sav
      inc fn
      lda fn
      jsr hex
      sta name1+$0e
      sta name2+$0e
      sty name1+$0f
      sty name2+$0f
      lda $01
      and #$10
      bne main
      jmp ($a000)

ton   lda $01
      and #$df
      sta $01
      rts

tof   lda $01
```

```
    ora #$20
    sta $01
    rts
lod
    lda #$01
    tax
    tay
    jsr $ffba
    lda #$00
    jsr $ffbd
    lda #$00
    sta $c0
    sta $93
    lda #$3c
    ldx #$03
    sta lda1
    stx ldah
    jsr $f7d7
    jmp $f84f
led
    lda #$01
    sta $c3
    lda #$08
    sta $c4
    lda $033f
    sbc $033d
    tax
    lda $0340
    sbc $033e
    tay
    clc
    txa
    adc $c3
    sta $ae
    tya
    adc $c4
    sta $af
    lda $c3
    sta $c1
    sta $ac
    lda $c4
    sta $c2
    sta $ad
    lda $033d
    sta lda1
    lda $033e
    sta ldah

    ldx #$07
p2   lda msg2,x
    and #$3f
```

```
        sta $0568,x
        dex
        bpl p2
        lda #$1b
        sta $d011
        jmp $f84f

dly     lda #$11
        ldx #$00
        ldy #$00
        beq st
g1      dex
        bne g1
        dey
        bne g1
st      sec
        sbc #$01
        sta $02
        lda #$df
        cmp $dc01
        beq ex
        lda $02
        bpl g1
ex      rts

sav     lda #$10
        jsr $ffbd
        ldx #$07
p3      lda msg3,x
        and #$3f
        sta $0568,x
        dex
        bpl p3
        lda #$1b
        sta $d011
        lda #$01
        ldx #$08
        ldy #$01
        jsr $ffba
        jsr $ffc0
        ldx #$01
        jsr $ffc9
        lda lda1
        jsr $ffa8
        lda ldah
        jsr $ffa8
savea   ldy #$00
        sei
        inc $01
        lda ($c1),y
        dec $01
```

```
        jsr $ffa8
        inc $d020
        dec $d020
        inc $c1
        bne s1
        inc $c2
s1      lda $c1
        cmp $ae
        lda $c2
        sbc $af
        bcc savea
        jsr $ffcc
        lda #$01
        jmp $ffc3

namer   ldy #$00
        ldx #$00
dname   lda $0341,y
        cmp #$20
        beq ok
        cmp #$2f
        bcc der
        cmp #$5a
        bcs der
        cmp #$40
        bcs ok
        cmp #$39
        bcs der
ok      sta name1,x
        sta name2,x
        inx
der     iny
        cpy #$0d
        bne dname
        rts

fn      .byte $00
lda1    .byte $00
ldah    .byte $00
name1   .text "h          h00"
name2   .text "f          f00"
msg2    .text "found   "
msg3    .text "saving  "

hextab  .text "0123456789abcdef"

hex     tax
        and #$0f
        tay
        lda hextab,y
        tay
```

```
    txa
    lsr a
    lsr a
    lsr a
    lsr a
    tax
    lda hextab,x
    rts

buffer .byte $00,$00,$00,$00
       .byte $00,$00,$00,$00
```

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

Permanent link:  
<https://codebase64.org/doku.php?id=base:rom-load>

Last update: **2015-04-17 04:33**

