

```

YSCROLL= $CF00 ; Vertical scroll table (moves bad lines)
STRETCH= $CF80 ; Stretch table
COLORS= $CE80 ; Table for background colors
YCOORD= $0380 ; Sprite y-positions (eight bytes)
HEIGHT= $0388 ; Sprite stretches (eight bytes)
YPOS= 52 ; Sprite y-coordinate
SPRCOL= 2 ; Sprite colors

```

```

*= $1000

```

```

SEI ; Disable interrupts
LDA #$7F
STA $DC0D ; Disable timer interrupts
LDA #<IRQ ; Our own interrupt handler
STA $0314
LDA #>IRQ
STA $0315
LDX #$3E ; We create a sprite to cassette

buffer
LOOP LDA SPRITE,X
STA $0340,X
DEX
BPL LOOP
LDX #7
LOOP2 LDA #$D ; Set the sprite image pointers
STA $07F8,X
LDA #SPRCOL ; Set sprite colors
STA $D027,X
DEX
BPL LOOP2
LDX #$26
LOOP3 LDA VIDEO,X ; Init VIC
STA $D000,X
DEX
BPL LOOP3
LDX #$7F ; Create the y-scroll table
LOOP4 TXA ; and clear the color table
AND #$07
ORA #$10 ; Non-blank screen
STA YSCROLL,X
LDA #$00
STA COLORS,X
DEX
BPL LOOP4
STA $3FFF
LDX #23 ; Create a color table

```

```

LOOP5          LDA BACK,X
               STA COLORS+8,X
               STA COLORS+32,X
               STA COLORS+56,X
               STA COLORS+80,X
               STA COLORS+96,X
               DEX
               BPL LOOP5
               JSR CHANGE          ; Init sprite sizes and y-
positions
               CLI                ; Enable interrupts
               RTS

IRQ           LDX #$01
               LDY #$08          ; 'normal' $D016
               NOP                ; Timing
               NOP
               NOP
               BIT $EA           ; (Add NOP's etc. for NTSC)
LOOP6         LDA YSCROLL-1,X    ; Move the screen (bad lines)
;5
               STA $D011
;4
               LDA COLORS,X      ; Load the background color
;4
               DEC $D016         ; Open the border
;6
               STA $D021         ; Set the background color
;4
               STY $D016         ; Screen to normal
;4
               LDA STRETCH,X     ; Stretch the sprites
;4
               STA $D017
;4
               EOR #$FF
;2
               STA $D017
;4
               INX                ; (Add NOP for NTSC      +2)
               INX                ; Increase counter
2
               BPL LOOP6        ; Loop 127 times
+ 3
               LDA #1           ; Ack the raster interrupt
=46

```

```

                                STA $D019
;+17(sprites)

                                JSR DOSTRETCH    ; New stretch
=63(whole)

                                JMP $EA31

SPRITE  BYTE 0,0,0,3,$FB,0,7,$7E           ; An Example sprite
        BYTE 0,$35,$DF,0,$1D,$77,0,$B7
        BYTE $5D,0,$BD,$83,$7E,$EF,1,$DE
        BYTE $BB,1,$78,$AE,3,$70,$EB,0
        BYTE 0,$BA,3,$60,$EE,3,$D8,$FB
        BYTE 2,$F6,$FE,$83,$BD,$9F,$BA,0
        BYTE $37,$EE,0,$3D,$FB,0,7,$7E
        BYTE 0,3,$DF,0,0,0,0

VIDEO   BYTE $E8,52,$20,52,$50,52,$80,52,$B0,52
        BYTE $E0,52,$10,52,$40,52,$C1,$18,52,-1,0,0
        BYTE $FF,8,$FF,$15,1,1,$FF,$FF,$FF,0,0,0,0,0,0,0,1,10
        ; Init values for VIC - sprites, interrupts, colors

BACK    BYTE 0,$B,$C,$F,1,$F,$C,$B       ; Example color bars
        BYTE 0,6,$E,$D,1,$D,$E,6
        BYTE 0,9,2,$A,1,$A,2,9

DOSTRETCH
                                LDX #31           ; Clear the table
                                LDA #0           ; (Unrolling will help the
speed,
LOOP7   STA STRETCH,X             ; because STA nnnn,X is 5
cycles
                                STA STRETCH+32,X ; and STA nnnx is only 4
cycles.)
                                STA STRETCH+64,X
                                STA STRETCH+96,X
                                DEX
                                BPL LOOP7
                                STA REMAIND+1   ; Clear the remainder
                                LDA #7
                                STA COUNTER+1   ; Init counter for 8 loops
                                LDA #$80
                                STA MASK+1     ; First sprite 7, mask is $80
COUNTER LDX #$00                 ; The argument is the counter
                                LDY YCOORD,X   ; y-position
                                LDA HEIGHT,X   ; Height of one line (5 bit
integer part)

```

```

                                STA ADD+1
                                LDX #20                                ; Handle 20 lines
LOOP8                            LDA STRETCH+2,Y
MASK                             ORA #$00
                                STA STRETCH+2,Y                    ; Set a one-bit
                                STY YADD+1
REMAIND                          LDA #0
                                AND #7                            ; Previous remainder
ADD                              ADC #0                            ; add to the height
                                STA REMAIND+1                        ; Save the new value
                                LSR
                                LSR
                                LSR
                                CLC                                ; Take the integer part
YADD                             ADC #0
                                TAY                                ; New value to y-register
                                DEX
                                BNE LOOP8
                                LSR MASK+1                          ; Use new mask
                                DEC COUNTER+1                       ; Next sprite
                                BPL COUNTER

CHANGE                           LDA #$00
                                ASL                                ; Sprite height changes with
                                ; 2x speed
                                AND #$3F
                                TAY                                ; 64 bytes long table
                                INC CHANGE+1                        ; Increase the counter
                                LDX #7                            ; Do eight sprites
LOOP9                            LDA SINUS,Y
                                LSR
                                LSR
                                CLC                                ; Use the same sinus as y-
                                ; data
                                ADC #8
                                STA HEIGHT,X                       ; Sprite height will be from
                                ; 1 to 3 lines
                                TYA
                                ADC #10                           ; Next sprite enlargement
                                ; will be 10 entries
                                AND #$3F                           ; from this
                                TAY
                                DEX
                                BPL LOOP9
                                LDX #7
                                LDA CHANGE+1
                                AND #$3F

```

