

Simple parallax shifting

by Achim

A parallax effect can be achieved by shifting bits and bytes of one char or a pattern of chars, e. g. 2x2 tiles, corresponding to the soft scroll registers (\$d016 & \$d011).



This effect can be seen in numerous games (e.g. Bounder, Parallax, Marauder etc.). Here's a small example for a 2x2 tile, hires.

Example 2x2 tile with chars A, B, C, and D.

```
AB
CD
```

To make this tile scroll left/right or up/down four shifting routines are necessary.

1. Shifting bits left/right

In mc-mode these loops have to be called twice per frame.

Scroll tile to the left

```

loop1:   ldx #$00
         lda charB,x           ;bit7 charB -> carry
         asl
         rol charA,x           ;carry -> bit0 charA
         rol charB,x           ;bit7 charA -> bit0 charB
         lda charC,x           ;same here with C and D
         asl
         rol charD,x
         rol charC,x
         inx
         cpx #$08
         bne loop1
         rts

```

Scroll tile to the right

```

loop2:   ldx #$00
         lda charA,x           ;bit0 charA -> carry
         lsr
         ror charB,x           ;carry -> bit7 charB

```

```

    ror charA,x      ;bit0 charB -> bit7 charA
    lda charC,x     ;same here with C and D
    lsr
    ror charD,x
    ror charC,x
    inx
    cpx #$08
    bne loop2
    rts

```

2. Shifting bytes up/down

Scroll tile up.

```

    lda charA      ;save byte0 charA
    pha
    lda charC     ;save byte0 charC
    pha
    ldx #$00
loop3a:  lda charA+1,x
        sta charA,x
        lda charC+1,x
        sta charC,x
        inx
        cpx #$07
        bne loop3a
        pla
        sta charA+7      ;byte0 charC -> byte7 charA
        pla
        sta charC+7     ;byte0 charA -> byte7 charC
        lda charB      ;same here with B and D
        pha
        lda charD
        pha
        ldx #$00
loop3b:  lda charB+1,x
        sta charB,x
        lda charD+1,x
        sta charD,x
        inx
        cpx #$07
        bne loop3b
        pla
        sta charB+7
        pla
        sta charD+7
        rts

```

Scroll tile down.

```
        lda charC+7
        pha
        lda charA+7
        pha
        ldx #$06
loop4a:   lda charC,x
        sta charC+1,x
        lda charA,x
        sta charA+1,x
        dex
        bpl loop4a
           pla
        sta charC
        pla
        sta charA
        lda charD+7
        pha
        lda charB+7
        pha
        ldx #$06
loop4b:   lda charD,x
        sta charD+1,x
        lda charB,x
        sta charB+1,x
        dex
        bpl loop4b
           pla
        sta charD
           pla
        sta charB
        rts
```

3. How to

These shifting routines can be used in different ways depending on what kind of parallax effect you want to achieve. A tile like this can now be put anywhere on the map. To make the tile look fixed while the background graphics are scrolling (e.g. Marauder) the main program has to call the inverse shifting routine:

background scrolling left > tile scrolling right (and vice versa)

Another very common effect is to make the tile scroll slower than the background (e.g. Parallax or Bouncer). In this case the inverse scroll direction has to be called every second frame.

Here's a tiny example with tiles in the background scrolled slower: [simpleparallax.zip](https://codebase64.org/files/simpleparallax.zip)

From:

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

Permanent link:

https://codebase64.org/doku.php?id=base:simple_parallax_shifting

Last update: **2016-11-27 02:17**

