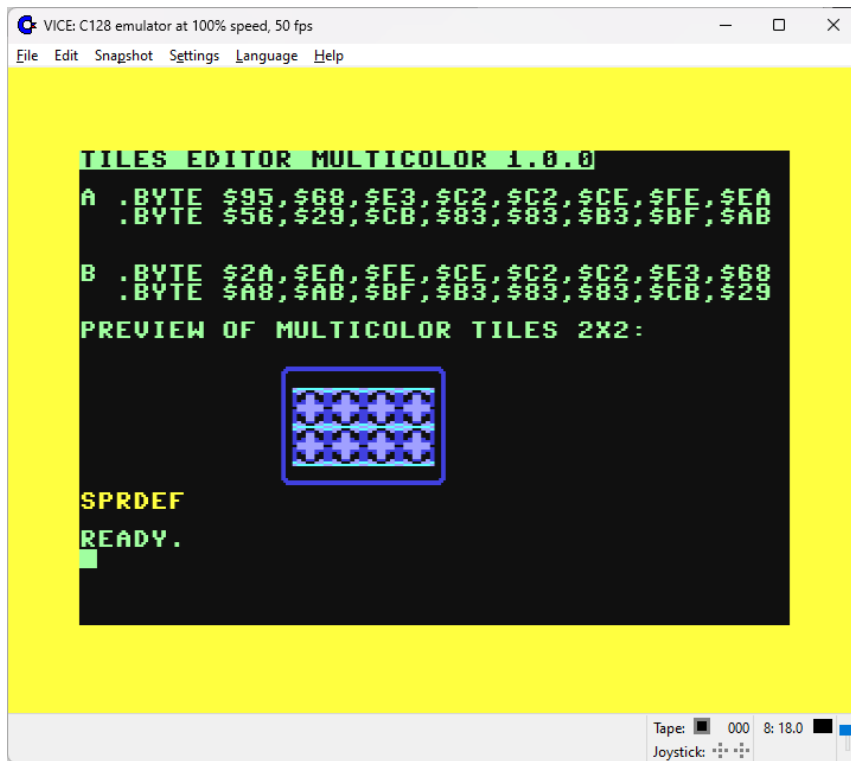# Tile Editor Multi Color Graphic Version 1.0.0

{updated to version 1.0.6A}, see page 3

With this little tool you can create tiles or any other graphic elements, with the size of 2x2 characters, using SPRDEF. SPRDEF, part of C128 OS, itself is for definition of Sprites, but you can use it literally for everything on top of that.

Preview is showing 1 Sprite {2x2 characters} and then 7 copies using other sprites.

This is done in assembly to speed things up. Copy progress is just way too slow in Basic.

This routine is located at $1500 and loaded with bload"sprcopy.bl"

It will load just once and then do the copy each time you start the main program in Basic.

```
. 01500   a9 d8      lda #$d8
. 01502   a0 00      ldy #$00
. 01504   a2 00      ldx #$00
. 01506   bd 00 0e   lda $0e00,x
. 01509   9d 40 0e   sta $0e40,x
. 0150c   9d 80 0e   sta $0e80,x
. 0150f   9d c0 0e   sta $0ec0,x
. 01512   9d 00 0f   sta $0f00,x
. 01515   9d 40 0f   sta $0f40,x
. 01518   9d 80 0f   sta $0f80,x
. 0151b   9d c0 0f   sta $0fc0,x
. 0151e   e8         inx
. 0151f   e0 63      cpx #$63
. 01521   d0 e3      bne $1506
. 01523   60         rts
```

{Copies Sprite one {actual tile} over to Sprites 2 to 8}
{$1502 is here not really needed but I am using it in ML Monitor because you never know}

$1500 is just adding a value to cell 1501 of #$d8 or dec 216. That is used to see if the routine was install and make sure not to load it over and over again each time main program is started to see the results.

One tile is 2x2 characters, starting left corner, multicolor {activate with M}

Which means that 1 pixel horizontally is 2 pixels wide as you can see in this screenshot from SPRDEF below. Vertical resolution is the same, meaning 8 pixels.

All counted together:
You are working with 2x2 Multi-Color characters = 8 {pixels wide} x 16 {pixels high}

4 Colors

1. Background {transparent, $d021,x or dec 53281,x}
2. Color 2 – standard color
3. Color 3 - multicolor 1
4. Color 4 - multicolor 2

SPRCOLOR is setting multi-color values for all sprites {graphic elements in this case},
and SPRITE command is setting regular color of each sprite {graphic elements in this case}
See Basic program for details.

You can use these graphic elements to create backgrounds in low-res mode of C128
(Graphic 3, x and Graphic 4, x, x}

An assembly routine to fill the first 12 lines of the screen {Graphic 4, split-screen gfx/text}, but it can be easily modified to do the same for the full screen multi-color mode {Graphic 3}, is on floppy.

It is preset to repeatedly draw 2 characters first line and 2 characters {tiles} the 2nd line.

However, with ease you can adapt it to draw 12 (or whatever you might need) individual lines.

And then, it also can be used for high resolution graphic since all you need to do is use Graphic 1 or Graphic 2 {see additional info} and not use Multi-Color. Everything else is the same.
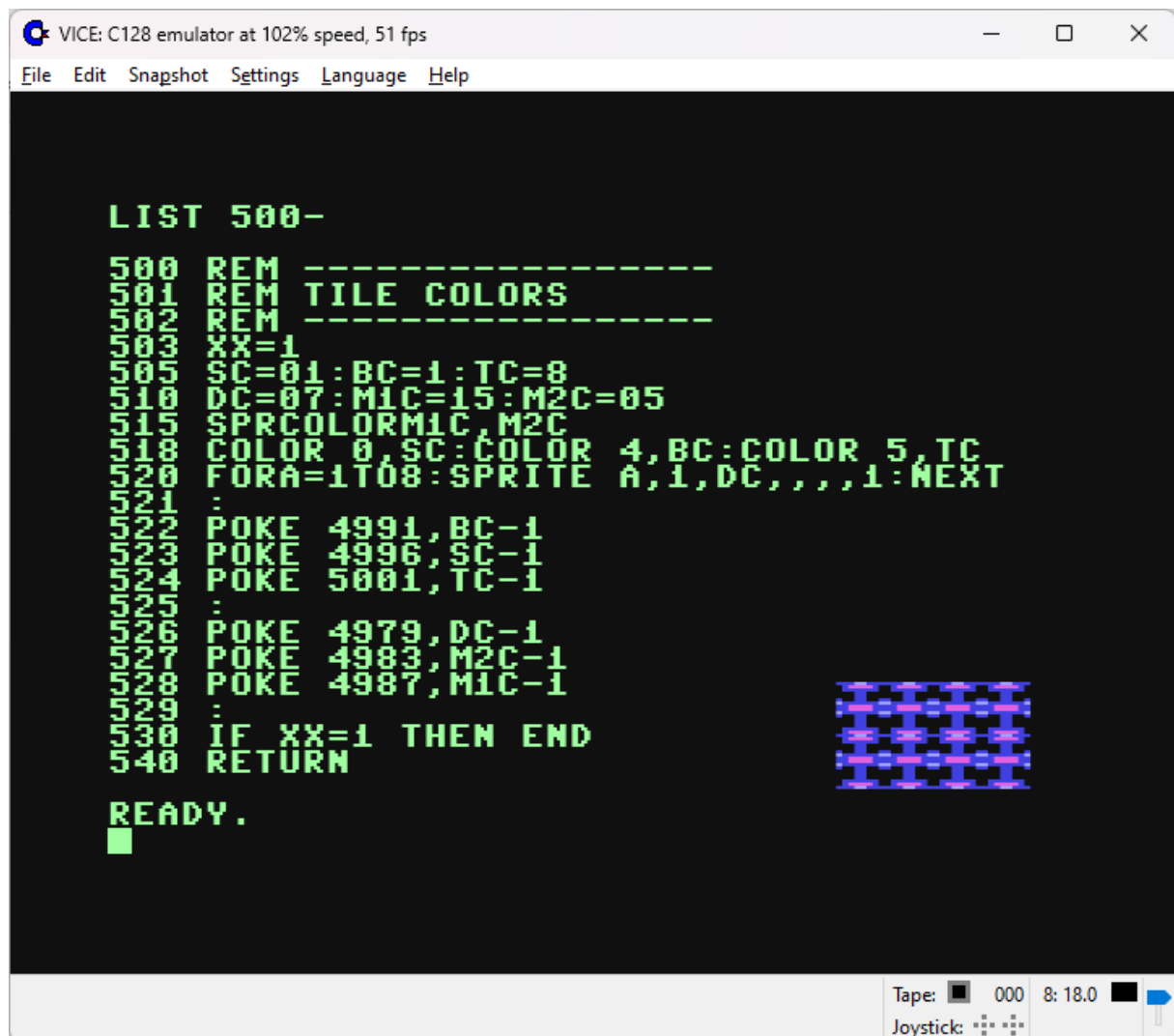Data export is working for single color and multi-color.

Lived.ch, 13.03.2024

Tile Editor Multi Color Graphic Version 1.0.6A

You can now list lines 500-, change values in line 505 and 510, then type run 500.
That will help you too choose the colors you want for your tiles.

Preview in Graphic mode?

Just type sys 4864 and see the tiles in Graphic 4 mode { split-screen }

Poke 4991, color #border color {BC, Color 4,x}

poke 4996, color #screen color {SC, Color 0,x}

poke 5001, color #text color {TC, Color 5,x}

Poke 4979, color #$84 {DC}

poke 4983, color #$85 {M2C}

poke 4987, color #$86 {M1C}

Note that Color 0,1 is equal to poke 53281,0 or lda #$00, sta $d0201!

Color start value is 1 and of poke and assembly is 0.

That's why it's poke sc-1, to match the values and have the same colors!