Graphics on the Commodore 64

Graphics on a 16kb Shoestring Budget

by

DeeKay/Crest

Part 1

Meet the Pixels

Basics

- PAL-VIC: 312 Rasterlines, no halfpictures (identical lines in both half-pictures)
- 4 Banks à 16KB, all displayed VIC-data needs to be within the same bank (Bitmap, Chars, Sprites)
- Rasterbeam goes linewise from the top left to the lower right, VIC also works that way
- 320x200 screen plus border (2 sprites=6 chars each on left/right, about 2 sprites upper/lower)

Basics

- Palette of 16 Colors, fixed
- 1 Char (Character) = 1 CPU-cycle
- No direct adressing of pixels

Some important VIC-Registers you should remember:

- $d020: Border color
- $d021: Screen color
- $d022/23: Multicolors
- $d016: X-shift Register
- $d012: Rasterbeam counter
- $d011: Good for a great many things (VIC-tricks)

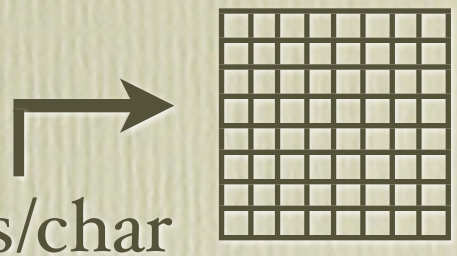
Lingo

- “Char”: One 8x8 pixel block (doesn’t matter if it’s actually char-mode!)
- “MSB”: Most Significant Bit: To cover overflow of an x/y value above $ff
- “Border”: The “outer safe frame” area surrounding the screen (made to compensate for various different TVs and monitors)
- “Koala”: synonymous for Multicolor Bitmap, named after a program called “Koala Painter”

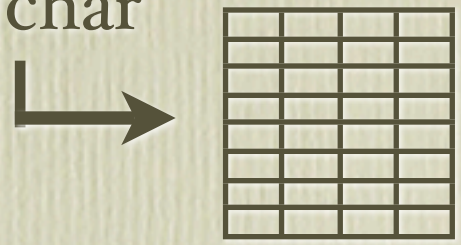
Displaying Graphics

- Standard Bitmap
- Character Set
- Sprites
- Advanced (but fairly useless) Stuff: Ghost-Byte, $d016

Graphic Modes



- Hires: square Pixels: two colors, 8x8 pixels/char



- Multicolor: 2 bpp, four colors : 4x8 pixels/char

  $d021: Bitpattern **00**

  $d022/Hi-Nibble (ScreenRAM): Bitpattern **01**

  $d023/Low-Nibble (ScreenRAM): Bitpattern **10**

  $d800 (”ColorRAM”): Bitpattern **11**

Standard Bitmap

- Hires 320*200, Multicolor 160*200
- Divided into 40*25 Chars (8*8 Hires-pixels),
- char-sequential: no linear adressing!
- 8000 Bytes = $2kb
- Fixed positions in RAM: $0000, $2000 etc.
- Changeable by bank-switching

Standard Bitmap Hires/ Multicolor

- Hires: 2 colors per char, colors are stored in ScreenRAM
- ScreenRAM: length $0400, fixed positions in active bank: $x000, $x400, $x800, $xc00 etc., colors are high and low nibble
- Multicolor: 4 colors per char, ScreenRAM + ColorRAM + Background ($d021)
- ColorRAM: length $0400, always on $d800, low-nibble only

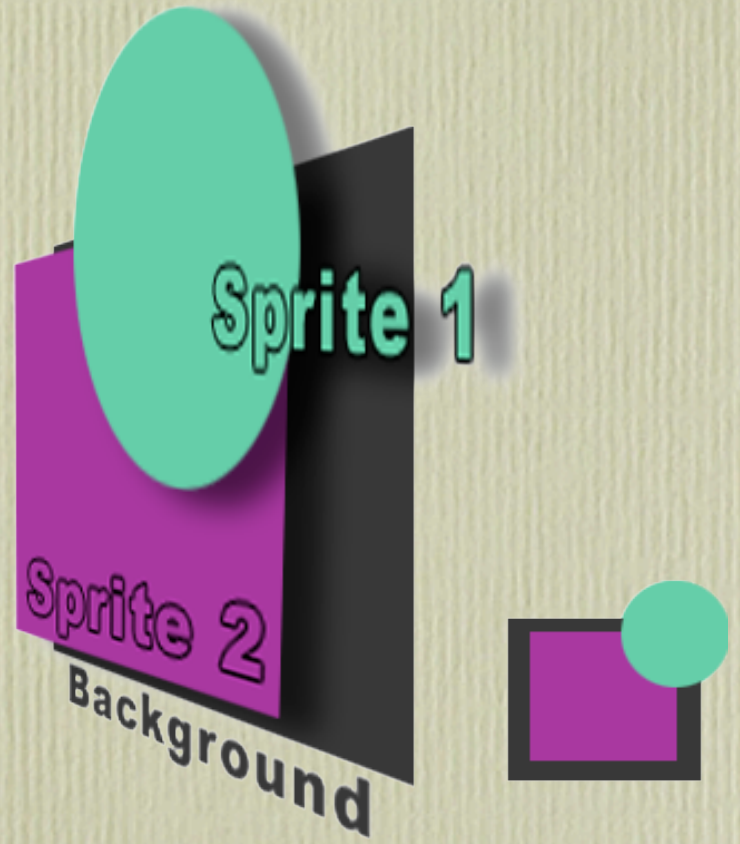
Sprites

- 24*21 pixels (hires) = 63 bytes
- sequential byte-order by line, no chars
- 8 sprites, can be multiplexed (horizontally no more than 8)
- Multicolor/Hires can be set individually per sprite
  Hires: one color plus transparent background
  Multicolor: 3 colors plus transp. background, 2 colors fixed for all sprites, one individual color per Sprite

Sprites

- Can be expanded to twice the width and/or height (pixels scale accordingly)
- Collision between sprites or sprites and chars can be detected
- Trickery! Sprite 1 can be behind Bitmap/Char, but *above* Sprite 2 which itself is above the bitmap. This essentially cuts a hole in Sprite 2 letting the bitmap shine through -> EOR-effects and more!

Sprite-"Cookiecutter"



Sprite 1 in front of Background and Sprite 2

Sprite 1 still in front of Sprite 2
(but behind Background!)

Sprite-Priority

- Sprites can be above or behind bitmap/chars
- Behind set bits in hires, in Multicolor only behind $d023 (or ScreenRAM Low-Nibble) & $d800 (ColorRAM)
- Tricks possible with covering

Charset

- 256 characters per char, 8 bytes per character, sequential store by 8*8 char
- Length: $0800
- Located on $x000, $x800, $xc00 of active bank
- Hires: One color out of 16 per char (ColorRAM)
- Multicolor: 2 fixed colors + $d021 + ColorRAM (only first 8 colors, second 8 are the same only in hires)

Charset

- Screen for charset: same as ScreenRAM in Bitmap, on $x400 boundaries
- New mode besides Multicolor/Hires: ECM: 64 Hires-Chars where each can have one of 4 different background colors (Multicolor-Registers, $d021-24). On top of that is each char's ColorRAM.

Making Graphics on the C64

- 16 Colors
- Many ways to go from black to white: Lots of possible transitions
- Seven Color-Pairs with same luminance for all colors plus black and white (a/c/e/5: special case!)
- sensible choice of colors for each char: choose colors that you can use often, don't use colors from the same pair if you're starved of colors

Making Graphics on the C64

- Don't choose the most often used color as $d021, choose the color that you can use as substitute most often: No black, no white, cannot be used as substitute or for Antialiasing. Good choice because color-less: One of the 3 greys
- Predominantly for Bitmap, but same goes for choice of Sprite- and Char-Multicolors

Old VIC special

- Very early C64s have a different VIC. Only five different luminances instead of nine
- Many colors are the same luminance -> Color-fades look worse
- Not very many in existence, so dont worry too much
- Ask Widdy! ;-)

Advanced c64-Graphics

- Most advanced graphicmodes can be reached through various tricks that change VIC-registers at the right time while the graphic is being displayed (c64 version of Pixelshaders! ;-)
- More colors or higher resolution, but hard to use in Demoparts with compute-intensive Effects, because it takes away rastertime while GFX is being displayed

Advanced c64-Graphics

- FLI: Flexible Line Interpreter, two new colors (ScreenRAM) every line instead of every 8th line, has FLI-Bug (solid light grey, 3 chars wide) at the left of the screen, either Hires (AFLI or Hi-FLI) or Multicolor; Editors: Blackmail/CPU Editor, FLIP

- IFLI: Interlaced FLI, same as FLI Multicolor, only interlaced (2 screens alternating) and with a 1-pixel x-shift every frame to give it a pseudo-hires look; Editors: Funpaint, Gunpaint

Advanced c64 Graphics

- Sprites overlay modes: SHF, SHIF (SHF interlace), SHF-XL: Hires-FLI-Bitmap with sprite-layer(s) over it for more colors; dont cover the whole screen, but are very high resolution; Editors: Crest SHF/SHIF/SHF-XL Editor
  Sprite underlay modes: UFLI and UIFLI (UFLI Interlace): X-expanded Hires-sprite-layer (twice as wide pixels) underneath the set pixels of a Hires-FLI-Bitmap (FLI every second line only); High resolution, fairly colorful, covers almost the full screen

Advanced c64-Graphics

- Border-GFX: VIC knows 23x38 char border for smooth scrolling. Fool the VIC by changing the registers into believing it switched on the border already when the rasterbeam reaches the border Open border has limited GFX-capabilities: Only $d021 & sprites, no chars, no standard bitmap, only one black repeating hires Ghostbyte in the upper/lower border (last Byte in the active Bank) plus one $d016-byte when you tear open $d021

Free Advanced Graphic Modes

- Advanced GFX-modes can be done without FLI, meaning the rastertime while the graphic is displayed can be used for other things.
- No FLI means less colors
- Sprite-overlay/underlay with Bitmap: Standard Bitmap (Multicolor or Hires), only Sprites need to be multiplexed every 21 Rasterlines
- Drazlace: Interlaced normal Bitmaps, with or without $d016 1-Pixel-x-shift

Linear Framebuffer

- How can I output my Code without the hassle of adressing the non-linear structure of c64-gfx?
- A nearly linear framebuffer can be emulated with various tricks, mostly in the end being some sort of a 4x4 pixel structure and 1/4 FLI.
- Some effects can be emulated with avoiding heavy calculation and relying on VIC-tricks instead, mostly also using Hi-Res graphics..

Examples of c64 Graphics

Hooray! Pwetty pictures time!

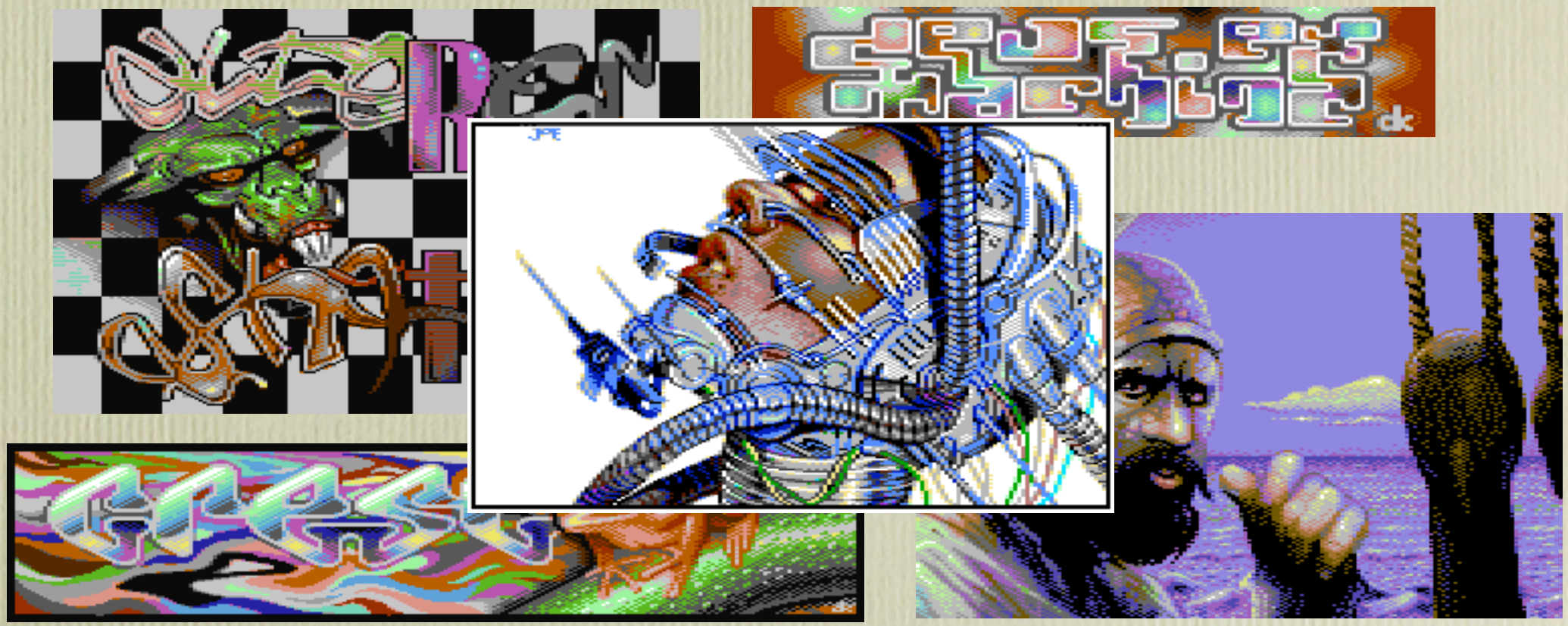
Examples

- Standard 4-color GFX (most often char)

Examples

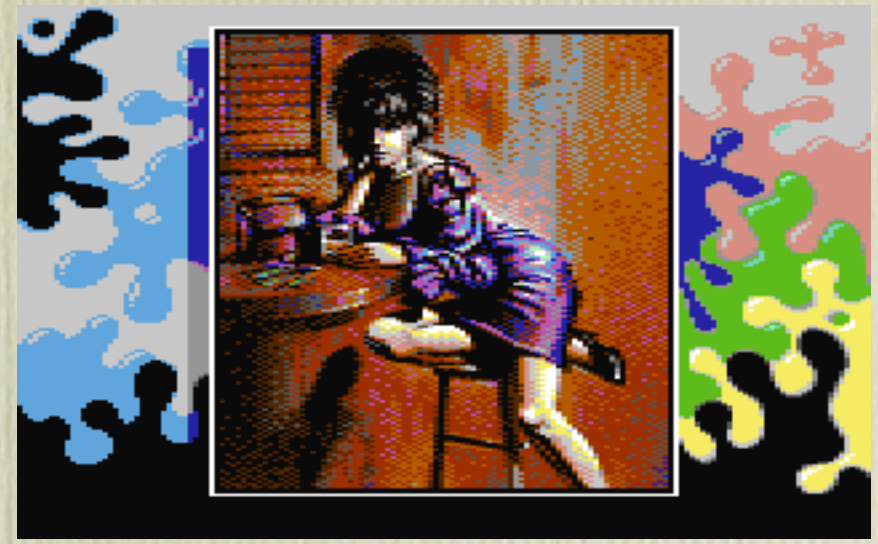
- Standard Multicolor (160x200, no interlace)

Examples



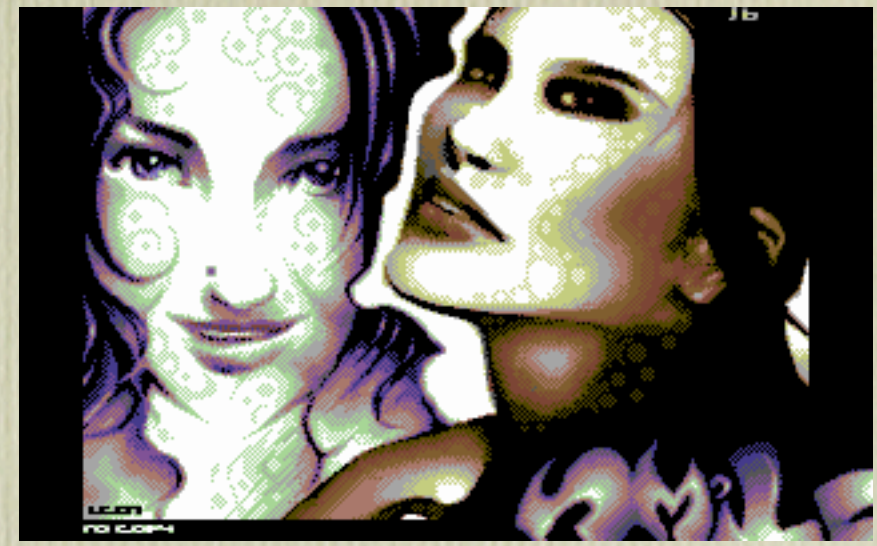
- Standard Multicolor, non-interlaced

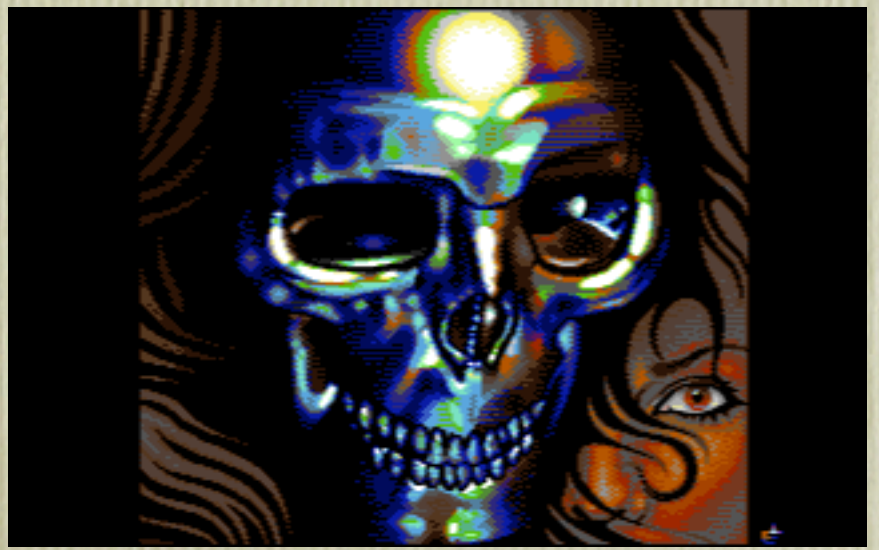
Examples



- FLI Multicolor, non-interlaced

Examples



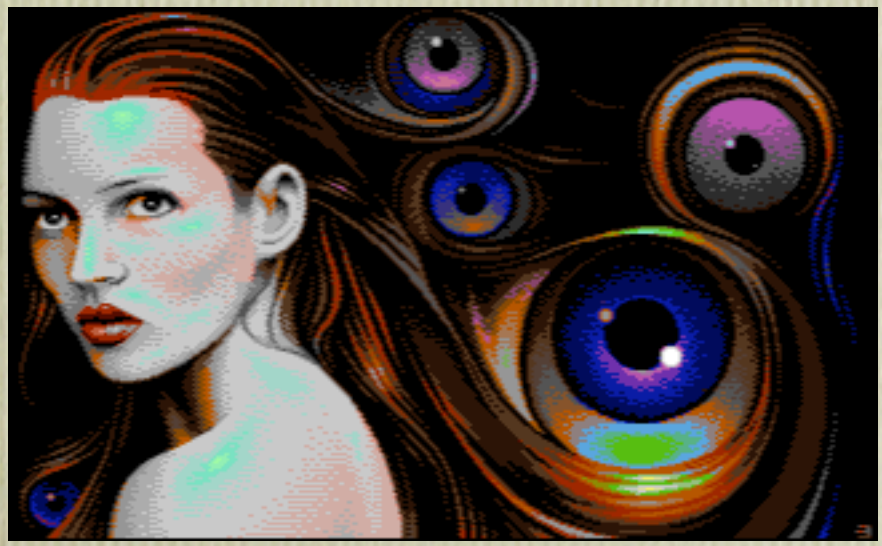


- Interlaced FLI

Examples

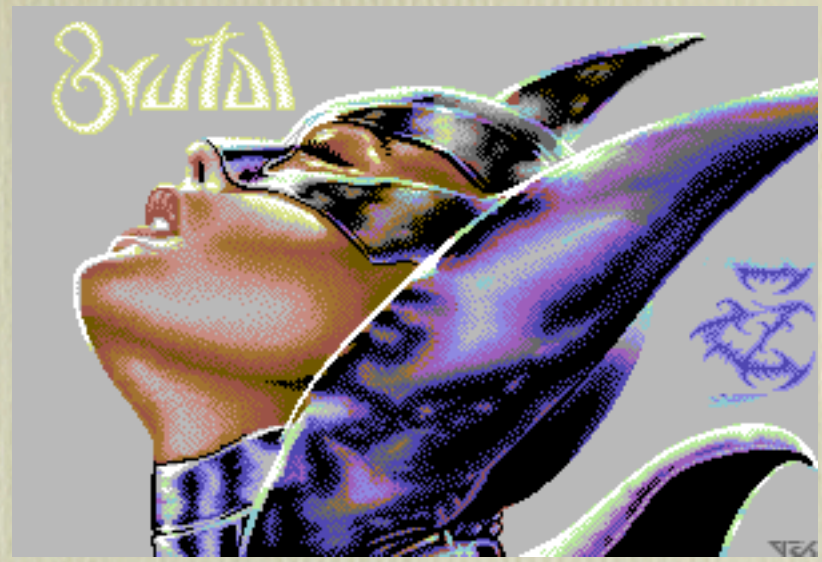
- Interlaced FLI

Examples



- Multicolor Interlace aka Drazlace (”free” advanced mode!)

Examples



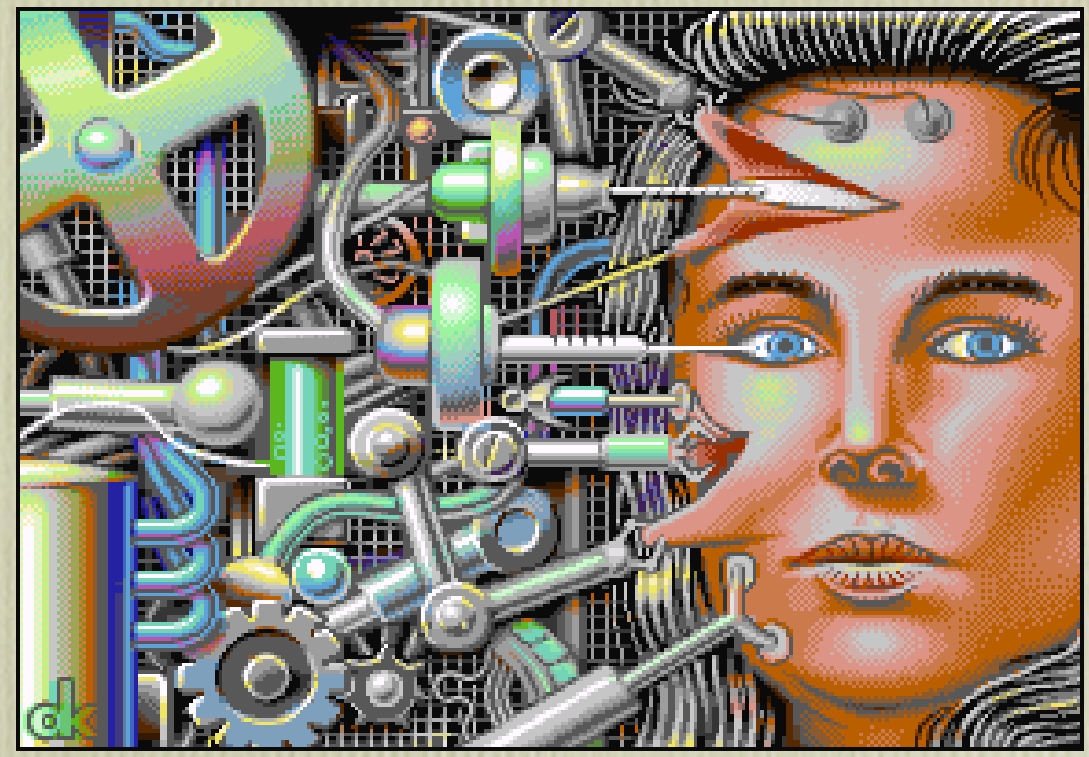
- UFLI (320x200, 1/2 Hires-FLI plus Sprite-Layer)

Examples

- Hires-FLI

- Super-Hires-Interlace FLI (SHIF, Interlaced Hires-FLI with 2 Hires Sprite-Layers, 96x112 pixels only)

Examples



- UIFLI: UFLI Interlace, the Crest of c64 Graphics: Very many colors, awesome transitions, much detail, covers whole screen

c64 Graphics with PC/Mac

Pixelling, converting and displaying

Areas of usage

PC/Mac can be a great tool

- Do Screendesigns for your coders!
- Display c64-Graphics with Xnview (Lin/Mac/Win), ACDsee+Plugins (Win), ShowVIC or various plugins on Amiga
- Do special graphics where editors do not exist on c64 on PC/Mac
- Pixel c64-Graphics with Timanthes or Elitepaint (Win)
- Use PC-Programs to do tedious/impossible jobs

Convert stuff

- Various converters available for all kinds of graphics modes (konv1.exe, BigVIC, Raydomat, ProjectOne, Timanthes etc)
- Be fair - Don't just convert readymade pictures by others and hand them in for the compo!
- Fix errors on the c64, converting is never as good as hand-pixelling

Making graphics in special modes

- Good coders WILL ask you to do graphics in 10x300 pixels with one new color every 5 characters and grey in all uneven lines - but only when the moon is full!
- Since the limitations are normally very uncommon, editors usually don't exist on c64
- If possible, use Photoshop and other tools to emulate these features - be creative!

Making graphics in special modes

- Example: A texture with 3 colors in every line

- Solution: Use Photoshop with 3 layers filled with only lines of one color, mask them and paint in the masks in b/w with a 2-pixel wide brush

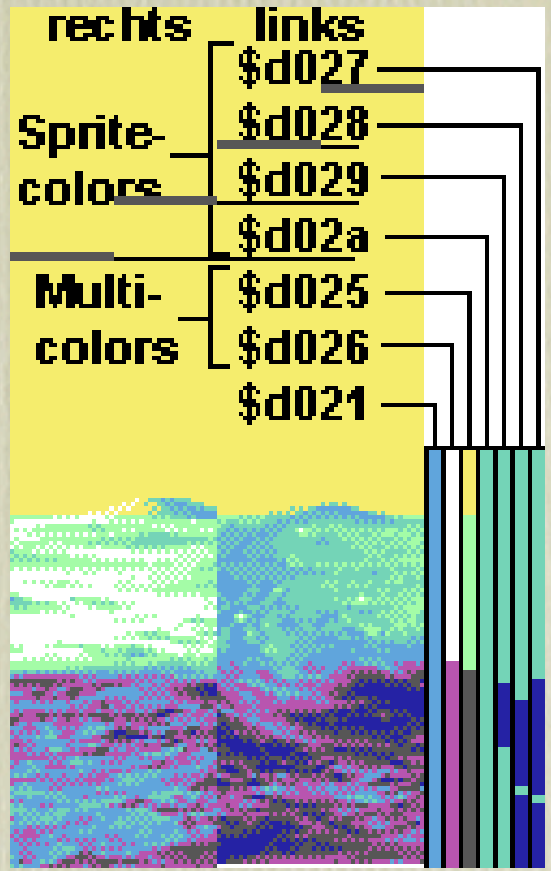
Making graphics in special modes

- Example 2: A sprite-logo (24 chars wide) with one new color every line

- Solution: Fill outlines with rasterbars, adjust remaining colors for needed additional fixes...

Making graphics in special modes

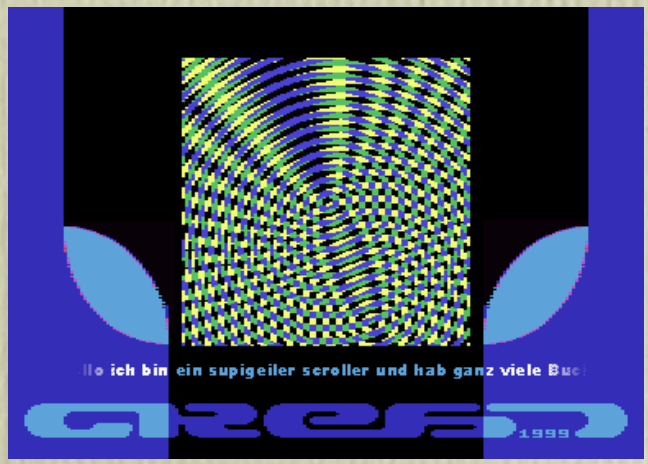
- Making Sideborder GFX in Photoshop for lack of a suitable Editor



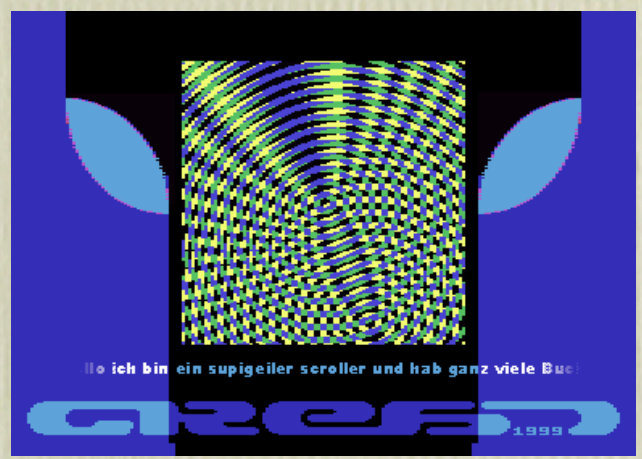
Screendesigns

Version 1

Version 2

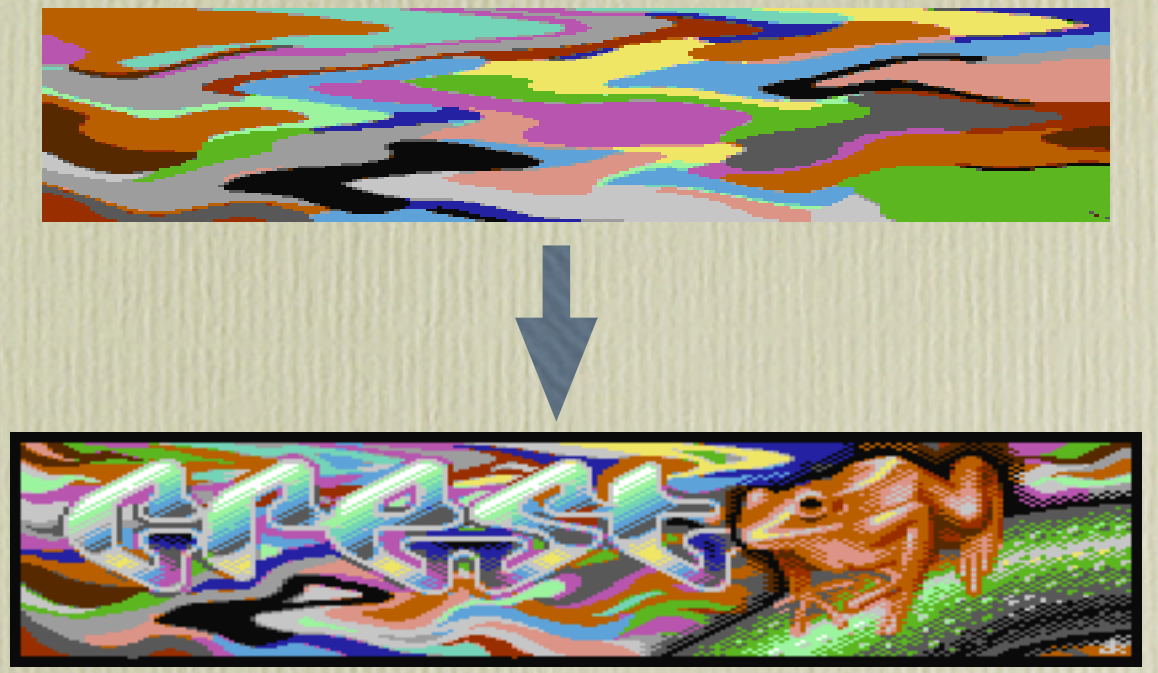


Version 3



Final part

Pattern-making



- Use Photoshop Functions to make background-patterns etc.

The End

Thank you for watching!

Glossary

- Multicolor FLI: also called “FLI”
- Multicolor Bitmap: also called “Koala”, “Amicapaint”, “Paintmagic”
- IFLI: Also “Funpaint”, “FLI Interlace”, FLInt or “Gunpaint”
- Hires-FLI: Also called “Hi-FLI” or “AFLI”