

# Demo Coding

This page is devoted specifically to demo effects and tutorials on how to make demos. More general graphics coding is instead covered in the [General VIC programming page](#).

To provide an example: Opening the border is commonly made in demo parts, but simply opening the border does not constitute a demo effect in itself (at least not during the last 20 years). Therefore you would find that on the General VIC programming page instead.

## General Information and Tutorials

- [An Introduction to Programming C-64 Demos](#) - Puterman/FLT. Extensive and easy to read tutorial aimed at beginners.
- [Demo world records and world firsts](#) - Who broke what record? Who invented effect X first? When? In what demo?

## Effects

Many demo effects depend on precise timing. Therefore it might be useful to have a look on the [Interrupts and Timing](#) page.

## Rasterbars

The most classic demo effect, apart from scrolling text.

- [Rasters - what they are and how to use them \(C=Hacking #3\)](#) - by Bruce Vrieling - (Note that there is a bug in the last example program that turns border black and white: the high and low bytes of "intcode" are reversed in the Ida instructions. Also note that on PAL machines the refresh rate is not 60hz, but 50 hz.)
- [Rasterbars source](#) - by Knoeki
- [Rasterbars 2 source](#) - by Bitbreaker
- [Rasterbars 3 source](#) - by Graham
- [Rasterbar Flasher source](#) - Flash Screen effect by Wozza/CygnusOz
- [Horizontal Raster Split](#) - Monte Carlos

## Scrolling text

Those scrolling texts that we all hate to love that we love to hate. Also see the [sprite section](#) for a sprite scroller.

- [Scroll text in common](#) - by Monte Carlos

- 1 char sized scrollers:
  - [Scroll text](#) - by Vai/Slash Design
  - [Scroll text](#) - By Richard Bayliss/TND
  - [Scroll text](#) - variable speed and direction, by Groepaz/Hitmen
- Zoom chars 8x:
  - [Char zoom](#) - by Conrad.
  - [Char zoom](#) - by Raf/Vulture Design
- Sprite scroller:
  - [Scrolltext using Sprites](#) - by Testicle
- Perspective Scroller
  - [Discofloor scroller as being used in Ächzeit](#) - by Bitbreaker/Oxyron^Arsenic^Nuance

## Swinging and tech-tech

- [TechTech](#) (or “wave”) - by Pasi 'Albert' Ojala (from “Demo corner” in C= Hacking 7).
- [TechTech](#) (using FLI routine) - by Compyx/Focus
- [Logo swing](#) - By Richard Bayliss

## 3D dot scroll

- [3D Dot Scroll](#) - by wegi /Black Sun/Samar/Fatum

## DYCP

- [DYCP](#) - Pasi 'Albert' Ojala (from “Demo Corner” in C= Hacking 6).

## DYSP

- [DYSP using sprite stretching](#) - by Compyx/Focus
- [DYSP using a cycle table](#) - by Compyx/Focus

## Plasma

- [ColorCyclePlasma](#) - By Cruzer
- [AFLI-Plasma](#) - by Testicle
- [Proportional-Charset-Noter with Plasma-Effect](#) - by Testicle
- [Copper Style FLI Plasma](#) - By Cruzer

## FPP (Flexible Pixel Position, aka Stretcher)

- [FLI-FPP-Scroller](#) - by Testicle

## Graphics Distortion

- [FLI Floffy](#) - By Cruzer
- [2nd Pixel FLI Distorter](#) - By Cruzer

## Fractals

- [Julia Fractal Morpher](#) - By dW

## Vectors

- [Drivecalc vectors](#) - 3D realtime filled vectors with 3D calculations done in the drive, by wegi /Black Sun/Samar/Fatum
- [Filling the vectors](#) - by Bitbreaker
- [Spritevectors](#) - by Bitbreaker

## Blending and Fading

- [Blend Charsets](#) by Chico /CIVITAS

## Starfields

- [8 Sprite starfield](#) - by Richard Bayliss
- [ROL Starfield using \\$d018](#) - by Richard Bayliss
- [Starfield](#) - by Fabrizio Stellato

## 2nd Line FLI

- [Twisters, x-rotators and waving carpets](#) - Bitbreaker/Oxyron^Arsenic^Nuance

## Fire Effects

- [4x4 charset fire with lots of colors](#) - Bitbreaker/Oxyron^Arsenic^Nuance

## Misc

- [Colour flashing \(notewriter style\)](#) - (Extended colour mode) by Richard Bayliss

- [just Animation](#) - explains the concept of how the animations in reanim8ed work. originally published in [Driven #31](#)

## Software screen modes for effects

- 16×16:
  - [16x16 char matrix](#) by Monte Carlos
  - [16x16 Matrix Scroll](#) by Chico /CIVITAS

## Optimization

- [Speedcode](#) by Cruzer/CML
- [Speeding up and optimising demo routines](#) - by conrad
- [Advanced optimizing](#) - by Bitbreaker/Oxyron/Nuance

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

Permanent link:  
[https://codebase64.org/doku.php?id=base:demo\\_programming](https://codebase64.org/doku.php?id=base:demo_programming)

Last update: **2019-01-22 16:07**

