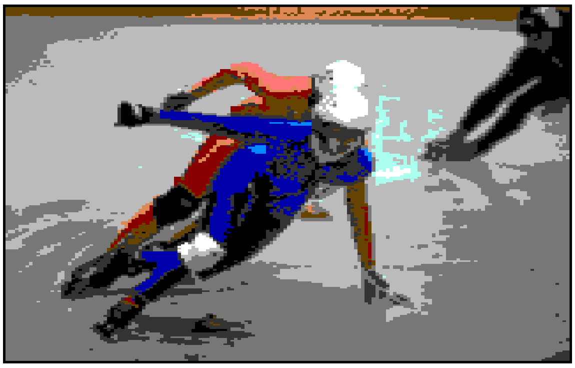
How to GoDot

By Arndt Dettke

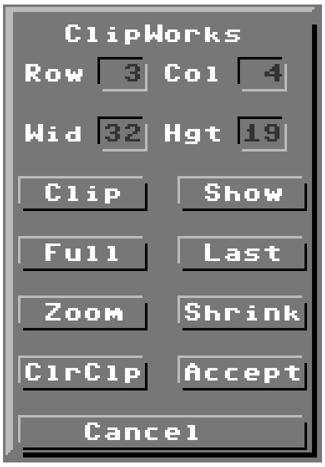
Working on clips (Part 1, Basics)

This time, we will use mod.ClipWorks, mod.FrameClip, mod.BalancingCol, ldr.4Bit&Mask and of course ldr.4BitGoDot. We're working on an image that can be downloaded from my site (see bottom of the article) called "icerunner. 4bt" – it's taken from the Windows clipart library and transported to our C=64 via GIF format. Also, at the end of this workshop you can see a list of all GoDot modules that relate to affecting clips.

Clips are rectangular areas of an image that you can manipulate individually. You may also save and load clips. **ClipWorks** is the central control module to work on clips.



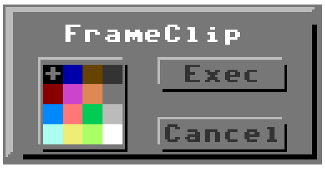
Now, let's start. Be sure to have ldr.4BitGoDot installed (it's the default loader, as you can see it in the upper left of GoDot's main screen). We first "***Load***" the image "icerunners.4bt" (and "***Replace***" the current image). After loading, we render the image to the graphics screen by clicking "***Display***". It's a nice image, but we will try to give it a little more artistic touch. What about this image with a frame in it, surrounded by a darker border? Well, it's easy!



Row 3 Col 4

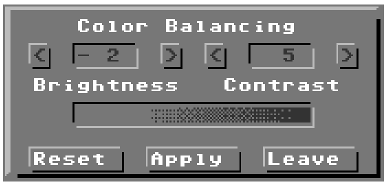
Wid 32 Hgt 19

The **ClipWorks** requester presents a facility to manually input values. Just click on the ***numbers*** at ***Row***, ***Column***, ***Width***, and ***Height***, and change them. Press RETURN when finished. If you want to visually determine a clip just click on "***Clip***", GoDot will show you the image (if you rendered it before!) There you click the upper left corner, go to the lower right and click again to set the clip. On SCPU's you better press RETURN instead of clicking the mouse, otherwise you can't set a decent clip. "***Show***" will bring up the graphics again and shows the current clip area blinking. "***Full***" resets the clip values to fullscreen. If you are in fullscreen mode (from main screen) you can activate the current clip if you click on "***Last***". "***Zoom***" and "***Shrink***" do what they are supposed to do, but in greyscale mode which leads to weird, perhaps unwanted colors. To take better advantage of it, zooming or better "scaling" will be covered in a future article. The "***ClrClp***" button gives you the opportunity to flood a clip by one of the C=64's colors. You choose a color similar to the way you do it in mod.FrameClip. Additionally, you may choose whether to clear either the "***Inside***" of a clip or everything else ("***Outside***").

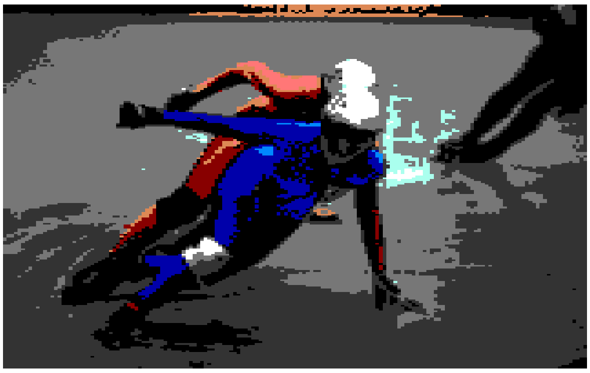


FrameClip lets you draw a colored border around the inner of a clip. The vertical lines are two pixels thick (so they can be recognized in multicolor mode), whereas the horizontal lines are just one pixel.

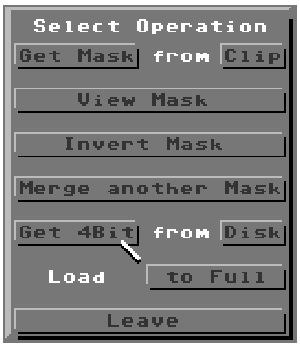
First, to darken the image (for the darker borders) you install **mod.BalancingCol** and "***Execute***" it. In the window that appears now you see two values (left for "***Brightness***" and right for "***Contrast***") which you can change by clicking the arrow buttons aside of them. Adjust them to –2 and 5 (which lowers the brightness by two steps and raise the contrast by five, so that the left and the right end of the pattern bar below the values show no pattern each) and "***Apply***" them. Render the image again ("***Display***") and see the difference.



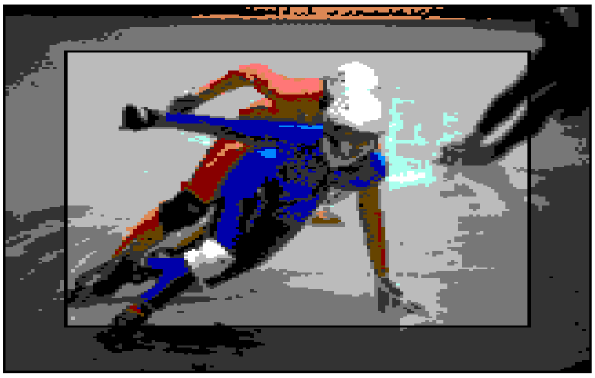
Now for the clip. Re-install mod.ClipWorks and "***Execute***" it. Set clip values of ***Wid*** 32, ***Hgt*** 19 ***Row*** 3, and ***Col*** 4, which describes a clip in the middle of the image, leaving three rows at the top and at the bottom and four columns left and right. Back on main screen, you notice that "**Exec Area**" has changed to "**Clip**" now. Next, this clip should be re-established to the old brightness values.



To achieve this most simply you install a new loader: **ldr.4Bit&Mask** and start it by clicking on the "***Load***" button. With this loader we will reload our "icerunners.4bt" image, but this time only to the place where the clip is. To do so, we have to coat the image with kind of a wax layer: click on the word "Disk" behind "Get Mask from". It will change to "***Clip***". Now click on "***Get Mask***" on the same line. You have created the wax coating now, and can have a look at it by clicking on "***View Mask***". The light grey rectangle is the wax layer, called a "**mask**". Unfortunately, it covers just the area it should leave uncovered! Well, go click "***Invert Mask***" and we're done correctly. Now click "***Get 4Bit***" and load the "icerunners.4bt" image again. When done, "***Display***" this latest stage of processing our image.



The framing we aimed for can be achieved by installing **mod.FrameClip** and executing it. It shows a little window in which you can select the color the frame should be drawn in. Choose black, "***Exec***", then switch back "Exec Area" to "***Full***" on main screen and "***Display***" again. This is all.



Looking good, eh?

Command history

(Load: 4BitGoDot)
Load Replace "icerunners.4bt"
Display
Inst: BalancingCol
Execute
Brightness: -2
Contrast: 5
Apply
(Display)
Inst: ClipWorks
Execute
Clip: 3, 4, 32, 19
Accept

```
(Exec Area: Clip)
Load: 4Bit&Mask
Load
Get Mask from: Clip
Get Mask
Invert Mask
(View Mask)
Get 4Bit “icerunners.4bt”
Leave
Inst: FrameClip
Execute
Select: black
Exec
Exec Area: Full
Display
```

GoDot’s modules supporting clips

mod..ClipInfo	info about clips on disk
mod.ClipWorks	covered in this article
mod.Flip&Mirror	turn clip contents
mod.FrameClip	draw frames around clips
mod.Scroll	scroll the image pixelwise
mod.Squeeze2Clip	scale image down to clip
mod.StretchClip	scale clip up to screen size
mod.Tile	scale image to a quarter
mod.TileClip	multiply a clip
mod.TurnClip	turn a clip by 90 degrees
ldr.4BitGoDot	load to/into clip
ldr.4Bit&Arith	load to/into clip
ldr.4Bit&Map	load to/into clip
ldr.4Bit&Mask	load only to clip
svr.4BitClip	save a clip
svr.PostScript	save a clip as EPS

Next issue will cover how to save and load clips. Many times when writing for an article I encounter missing features or also bugs in the modules I’m talking about. These will always be updated for that particular article and can then be downloaded from my site (http://www.godot64.de/download). The image material used in an article can also be grabbed from there (in subdirectory /workshops).

Have fun using GoDot!