English Amiga Board Mobile

Home Today's Posts

Coders Heaven>Raw Interleaved Graphics

Hungry Horace 23:56 28 January 2008

hello

i am looking to code a small program that will load, (ie. draw) raw interleaved graphics files.

Unfortunately i have little idea as to how the format works, and how it is structured. I am told it is related to bitplanes, and that the data in bits will be used to turn the data back into its palette related data... however, even a look into the workings of bitplanes, does nothing to help me in knowing the difference between raw and raw interleaved structures.

ideally, i want to simply read through the data in sequence "drawing" my picture along the way, so it is only really a matter of interpretting the data and turning it back into a colour reference. The size of the final picture is already locked by the source data size, so this saves some bother here.

Perhaps someone kind enough could please explain a simple way of me doing this, that can fairly easily be translated into AMOS code.

thanks in advance for any help.

[Rep]

Photon 00:39 29 January 2008

A non-IL picture has all lines of the first bitplane, then all lines of the second bitplane, up to the picture's depth. Set first bitplane pointer to start of pic, second bitplane to start of pic+width/8*height, etc., with modulo 0 (normally).

An IL picture have all bitplanes after each other, line by line. So the first wid/8 bytes would be line 0 of the first bitplane, the next wid/8 bytes would be line 0 of second bitplane, up to the picture bitplane count. This is repeated for line 1, 2, 3... etc up to the last line of the pic.

This means bitplane pointers should be set up to point to start, start+width/8, start+width/8*2, etc, and modulo should be (bitdepth-1)*wid/8. So that when all the bitplane pointer have incremented from showing the current line, the "rest" should be added to skip to the start of "their" "bitplane-line".

Sometimes converters allow you to put the palette before or after your pic.

Edit: I should have answered the post, and not a few sentences in the chat :-)
To make such files, here's an example:
Width=16
Height=5000 :-)
Depth=4

Each bitplane would be two bytes wide, and there would be 4 of them to describe the top line of the picture. Directly after this comes the 4 words of the second line, etc. Until you get a humongous 2*4*5000=40000 byte chunk. Add 16 words of palette if needed.:-)

[Rep]

Hungry Horace 02:53 30 January 2008

hmmm... cheers photon, i think i understand it but i'm having real trouble implementing it into basic code. :-)

firstly, am i right in thinking that, when viewed in it untouched form, the data would read like this? (example is a 16 wide graphic again)

Code:

```
|A| 00    00   |B|    11
|A| 00    00   |B|    00
...
etc etc
```

Assuming that is correct, i think i have found it quite easy to make sure i'm peeking at the right data each time, even using wider graphics.

the trouble is then turning my found data for line 1 (00,11,22,33 for each bpl in this case) back into a stream of 0-15 numbers which can be used to "draw" the strip back onto the first line (y=0) of the screen.

i admit i'm probably confusing myself more than i need to, but i've never really understood bitmap-layers properly anyway! (i guess now is my chance to learn :-))
[Rep]

Photon 01:44 31 January 2008
EDIT:

Agh, sloppy editing. The four lines starting with "Word" are now correct :PPP

OK, /me is AMOS noob, but here's a try... :-)

Make an array (DIM?) of 16 color numbers.
Make an array of 4 words.

To read the first 16 pixels, read the first 4 words into the Word-array

```
LOOP FROM 15 DOWN TO 0 <--loop counter is pixel position from left (15) to 0 (right)
{
Word 0: If any of the bits #15 thru 0 is a "1", add 1 to color number (15 down to 0)
Word 1: If any of the bits #15 thru 0 is a "1", add 2 to color number (15 down to 0)
Word 2: If any of the bits #15 thru 0 is a "1", add 4 to color number (15 down to 0)
Word 3: If any of the bits #15 thru 0 is a "1", add 8 to color number (15 down to 0)
}
```

The color number array now contains 16 color numbers. To go to the next line in a 16px wide picture, simply loop all except the "make array" statements as many times as the picture is high, in pixels.
[Rep]

Photon 01:53 31 January 2008
NOTE: above post corrected. Bah. :-)
[Rep]

Hungry Horace 02:06 31 January 2008
yes, that is -very- understandable, thank you Photon for being kind enough to help out someone finding his feet.

Although your AMOS code is terrible, it's nice and generic, and i'm fairly sure i'll have it working soon :-)
[Rep]

Photon 20:25 31 January 2008
If there is no instruction in AMOS to read the bits of a word, you can calculate 2^loopcount (2 raised to the power of..) and AND each word with this. If the result is non-zero, add to the pixelvalue, otherwise skip to next.

Something like "if WordArray[0] AND (2^loopcount) then pixelcolor=pixelcolor+1" ?

[Rep]

Hungry Horace 20:39 31 January 2008

hmm.. well there isnt, except a bin$ command, which i chop the bits out of

it's not nice but it works.

it is however pretty slow at grawing the graphics. i might try your method and see if it helps, as its likely to be this calcuation slowing things down.

i would consider using a long-winded method of chopping up the raw file, and then converting to IFF, in order to gain speed..... the only trouble is, i cant find (been trawling aminet all day) a single command-line RAW->IFF converter.

anyone know of any?

[Rep]

Photon 22:07 31 January 2008

2 converters in the zone :-)

[Rep]

Hungry Horace 22:10 31 January 2008

> *Originally Posted by **Hungry Horace**:*
> *the only trouble is, i cant find a single **command-line** RAW->IFF converter.*

thanks for trying though mate, you've already been of help :-)

i think gfxcon should be able to do it, if i can work out the right parameters... but currently i cant get them right!

http://aminet.net/package/gfx/conv/gfxcon

edit (again): maybe 'RGB-Raw' doesnt include interleaved :-)

[Rep]

Photon 23:43 31 January 2008

I really suck in this thread, I guess it's so different from what I usually do on the Amiga. :-)

I never blame my mistakes on anything, but recent tragic events seem to make me brood about them and lose attention.

[Rep]

Hungry Horace 00:15 01 February 2008

> *Originally Posted by **Photon**:*
> *I really suck in this thread, I guess it's so different from what I usually do on the Amiga. :-)*

:-) you've been of great help mate... without, i wouldnt be able to do anything - at least now i can use my slower AMOS routine until either it's improved, or i can find a command-line alternative.

dont get yourself down too much.

[Rep]

Up

User Name

Password □ Log in

Page generated in **0.14029** seconds with **11** queries