Subject: Re: Inverse palette conversion algorithm...
Posted by mr_ped on Fri, 07 Apr 2006 11:16:43 GMT
View Forum Message <> Reply to Message

1) why to use indexed color? (ok, sometimes it's still really needed, but usually 32b ARGB is the way to go)
2) so you speeded up 20 times code which is run single time during init and takes under 100ms even with brute-force algorithm, and you needed 3 days to do that.
2 - related) in case you generate the conversion table more than one time, there's something fishy about that. (and looks like you can speed up the thing much more just by omiting multiple initialization of the same table)

While I like the actual trick and will probably check it more deeply to understand fully your approarch, I'm not sure it was worth of the effort in "economical" sense.
Than again, if you enjoyed it, it was probably worth of it anyway.

Ad my trick:
Beware it will stop to work if you get negative numbers in ((4-RASTER_SHIFT_X)*2) formula (i.e. RASTER_MAP_X is under 16).

(Also using signed "int" is not really safe when working with ARGB color channels, because the ">>" will be compiled as SAL, not SHL. As long as you are sure the MSb is zero it doesn't matter, but once you start to use alpha channel or "signed short" on 16bpp colors you are asking for trouble ... usually it's best to keep colors and channels in unsigned byte/word/dword)