

---

Subject: Re: Porting SystemDraw to Frambuffer  
Posted by [mirek](#) on Fri, 04 Sep 2009 16:53:20 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

kohait00 wrote on Fri, 04 September 2009 07:55hi mirek

the point is that we \*dont\* use X11, so we work on bare /dev/fb0 and drivers (/dev/input/\* or dev/mouse and even tslib stuff for touchscreen) to get that working..

I am asking because of development/debugging environment...

Quote:

you "simply" open /dev/fb0, but you then mmap it to your userspace and can then start painting on it (some more details on that needed). to find out mode or to set it, to find out strides, color width and the like there is this fbinfo struct (donno the name exactly, must look it up also) and according to that, BufferPainter needs to be set up. So maybe a wrapper class?

I think you are going to do double-buffering anyway, so ImageBuffer is just fine for this (as you can solve the format difference later...)

Quote:

where a SystemDraw is instantiated and drawn to GDC and passed all the events to.

It is during handling of WM\_PAINT (or expose).

Quote:

Where is that point exactly, i could find it seen thing like CreateWindow and CreateWindowEx, but the rest.... thats kind of specific.. i wanted to know where the start point for Upp actually begins, where OS things switch /hand over to Upp stuff/interface.

as a startpoint we might want to take a look at the nanox project in blackfin uClinux trunk, here is the start point maybe, how to use fb0 (in a simple way, nanox is smalles X11, but we might want to have no X11 at all.

svn: blackfin.uclinux.org, svn of uClinux distro  
trunk/user/microwindow/src/nanox/clientfb.c  
dig for /dev/fb0

Well, thinking about it, I guess nice and simple solution is to provide some abstraction level...

I mean, a class that abstractly represents sort of framebuffer and mouse and keyboard input and then build the whole thing above it.

For development purposes, we can use normal single window to emulate this.

Quote:  
another Point: SDL

i saw that SDL is somewhat supported, how's that?

Not really, the only support is that it comes with mingw version.

Quote:  
it can use bare fb0 and also has its layer to the input driver stuff, including touch screen. does Upp support SDL natively?? meaning without any GTK stuff and the like? or even in windows a native SDL?

No. But I guess we can use SDL as inspiration for above abstraction layer...

Mirek

---