

# Lazy Foo' Productions

[News](#)[FAQs](#)[Games](#)[Tutorials](#)[Articles](#)[Contact](#)[Donations](#)**Calcium Phosphate**

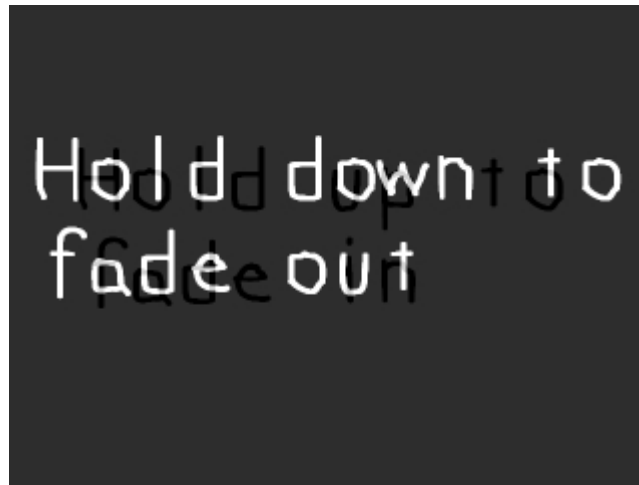
Powders, Solids, Coatings for Dental,  
Medical, Laboratory Uses

**Object COBOL Transition**

Upgrade your COBOL applications using  
Micro Focus Server Express

Ads by Google

## Alpha Blending



Last Updated 11/17/07

Alpha blending allows you to have have surfaces with different levels of transparency. In this tutorial we blit two surfaces with the front one being able to fade in/out by changing its alpha value.

```
//The frame rate  
const int FRAMES_PER_SECOND = 20;
```

SDL's alpha transparency is notoriously slow. That's because it's not hardware accelerated. Stick what ever graphic's card in your system it won't make much of a difference.

I just wanted to point out that if you notice SDL being really slow in this demo, it may be because the frame rate is capped at 20 fps.

If you want hardware accelerated graphics, consider moving on to OpenGL. It works very well with SDL.

```
//Quit flag  
bool quit = false;
```

```
//The front surface alpha value
int alpha = SDL_ALPHA_OPAQUE;

//The frame rate regulator
Timer fps;

//Initialize
if( init() == false )
{
    return 1;
}

//Load the files
if( load_files() == false )
{
    return 1;
}
```

Near the top of our main function we declare "alpha" to hold the alpha value of our front surface.

```
//Get the keystates
Uint8 *keystates = SDL_GetKeyState( NULL );

//If up is pressed
if( keystates[ SDLK_UP ] )
{
    //If alpha is not at maximum
    if( alpha < SDL_ALPHA_OPAQUE )
    {
        //Increase alpha
        alpha += 5;
    }
}

//If down is pressed
if( keystates[ SDLK_DOWN ] )
{
    //If alpha is not at minimum
    if( alpha > SDL_ALPHA_TRANSPARENT )
    {
        //Decrease alpha
        alpha -= 5;
    }
}
```

Here's where we play with our alpha value.

Alpha goes from 0 to 255 like Red, Green, or Blue. Alpha at 0 is completely transparent and equal to `SDL_ALPHA_TRANSPARENT`. Alpha at 255 is completely opaque and equal to `SDL_ALPHA_OPAQUE`.

In this piece of code we get the key states, then increase alpha when up is pressed and decrease alpha when down is pressed.

```
//Set surface alpha
SDL_SetAlpha( front, SDL_SRCALPHA, alpha );

//Apply the back
apply_surface( 0, 0, back, screen );

//Apply the front
apply_surface( 0, 0, front, screen );

//Update the screen
if( SDL_Flip( screen ) == -1 )
{
    return 1;
}
```

Here we apply the alpha value to the front surface using `SDL_SetAlpha()`. Now when the front surface is blitted, it will have the transparency we set.

Then we blit back surface, then the front surface with the alpha value we gave it and finally we update the screen.

Download the media and source code for this tutorial [here](#).

**Object COBOL Transition**

Upgrade your COBOL applications using  
Micro Focus Server Express

**Information on Alpha-1**

Check here for all the information on Alpha-1  
Antitrypsin Deficiency

Ads by Google

[News](#)   [FAQs](#)   [Games](#)   [Tutorials](#)   [Articles](#)   [Contact](#)   [Donations](#)

Copyright Lazy Foo' Productions 2004-2008