Color Palette Swapper Shader

This document will explain what is and how to properly set up Color Palette Swapper Shader.

Created by Hyago Pinheiro.
You can find the complete Unity package asset here.

What is a Palette Swap?

A palette is a texture containing only one pixel horizontally and multiple pixels vertically representing all colors used to create an image.

In the context of game development, a palette swap is the process of switching an image's set of colors (commonly known as *color palette*) by another one. This allows the same image to have multiple color variations. For a game, those variations may represent a increase of difficulty, a new player ability, a second character or any other visible gameplay feature.

Every image has a color palette. In our case, the palette contains only the colors we want to swap in the image and each color takes only one pixel size in the texture. Then, each palette is a very, very small texture file. As an example, for the image below a color palette containing only 4 colors are necessary because we want to swap only those 4 colors.

Figure 1 *- Mega-Woman with its original color palette.*

The palette color in Figure 1 has been increased to better visualise it. If we swap the color palettes, the whole image changes as well.

Figure 2 *- Mega-Woman with new color palettes.*

Note that the white color (for the eyeball) in the Figure 1 and 2 is not present in the palette. This is proposital since we don't want the eyeball color to ever change.

This simple technique is very common in the game development industry even today because allows us to save both computer memory and development time.

How to setup?

On Unity, the graphic element used is a texture or sprite. Using a special shader, one sprite can be used to create many distincts gameobjects in the same scene using differents palettes.

In this package, there are two shaders able to swap palettes. They are based on the default Unity Sprite shader, with all its original settings available. If your 2D game interacts with light, there is a variant of each shader based on the Unity Diffuse Sprite shader as well. In total, there are 4 shaders available to use and two script components to control them during gameplay.

The first shader is named **Palette Swap** and it's indicated if you just want to swap a palette by another one very quickly. You just need two textures: the original palette and a desired palette. The next figure shows two palettes:

Figure 3 *- Original and secondary color palette.*

Import the palettes textures into your Unity project. In Sprite inspector window, it’s important to set **Compression** to **None** and set **Filter Mode** to **Point**. This makes sure that no color noise due to compression will affect the color swap process.

Create a new material and select **Sprites/Palette Swap** or **Sprites/Palette Swap Diffuse** shader. Choose the diffuse shader if you want to enable light interaction. After that, you gonna see the inspector like the next figure depending witch shader you chose.

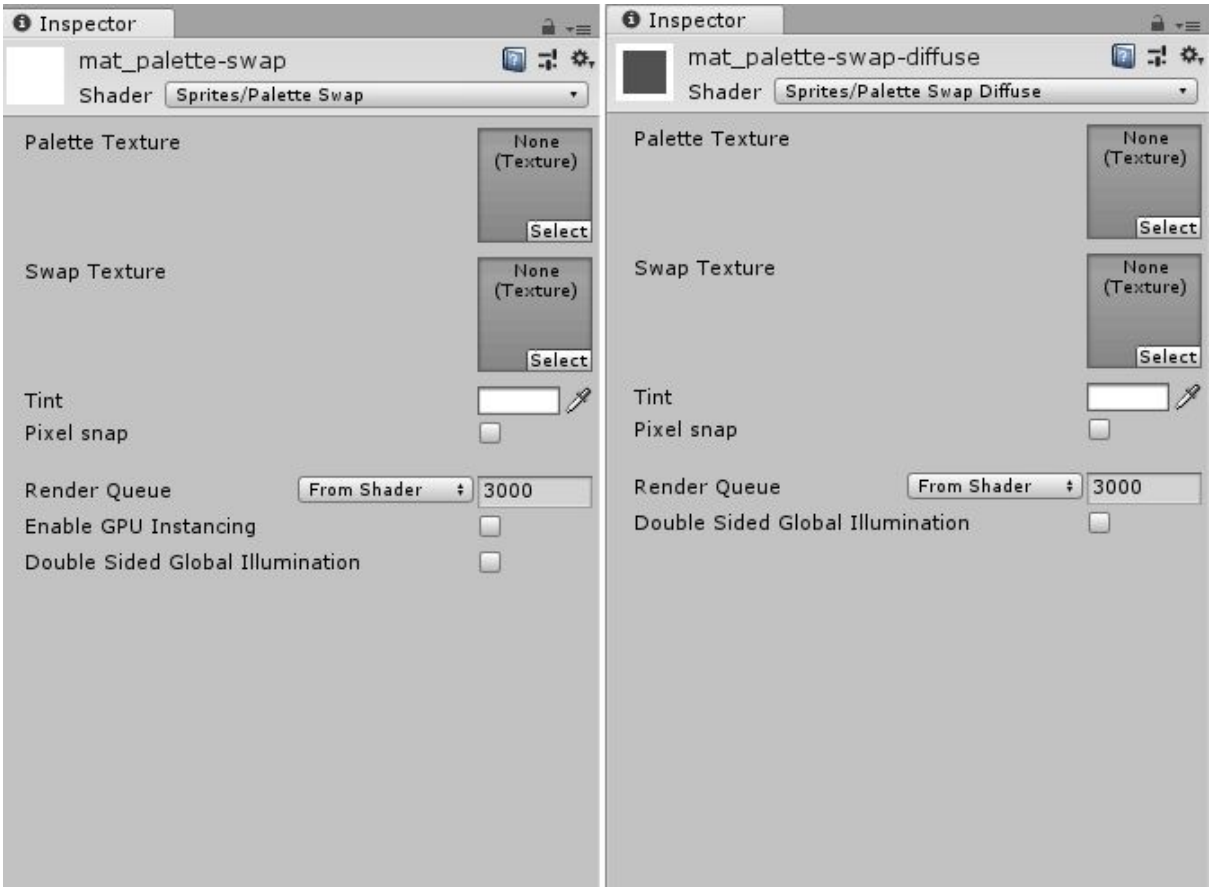

Figure 4 *- Palette Swap (left) and Palette Swap Diffuse shader inspector.*

After that, you must link the previous material you just created with the *Sprite Renderer* component of your gameobject. Now, put the original palette texture into the **Palette Texture** property and any secondary palette inside **Swap Texture** property. You should be able to see the sprite gameobject changes its colors on the scene right away.

Now if you want to swap colors during gameplay, you should attach a script component called **Palette Swap Controller** into the gameobject with the *Sprite Renderer* component. Set the public properties from this script on the inspector and, on another script, call the public *Next Palette, Previous Palette* or *Original Palette* functions. If you want to just test those feature you can use the **Palette Input Swapper** script provided in the package. Now set the inputs button names and press them during gameplay.

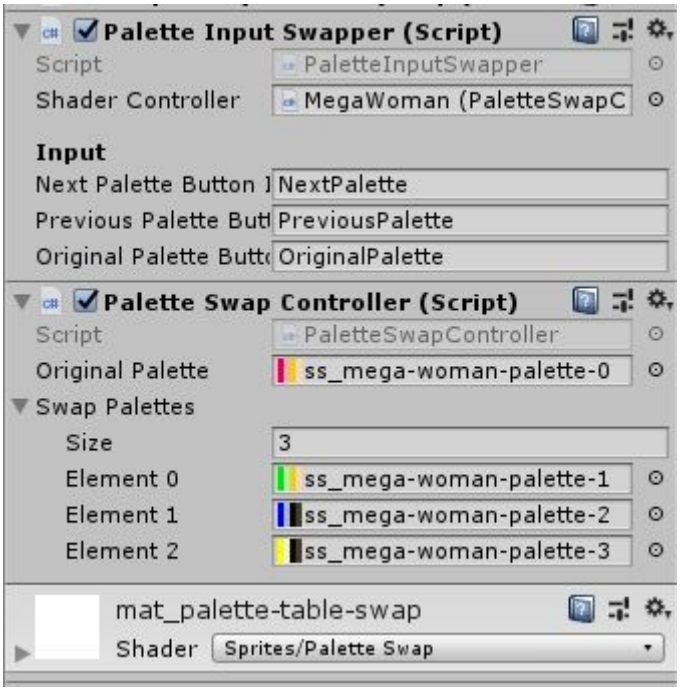

Figure 5 *- Palette Input Swapper and Palette Swap Controller components inspector.*

If your gameobject animates the sprite frames with the *Animator* component, everything will work as expected.

Now, if your game has a character who needs to swap colors multiple times using multiple palettes, you may use a palette table. A palette table is just an image where each palette is stacked in top of each other, like the next image:

Figure 6 *- Palette Table for Mega Woman.*

Each palette should have 1 pixel height and the original palette should be always the first one (from bottom to top). The next palette will be in the next row and so on.

Create a new material and select **Sprites/Palette Table Swap** or **Sprites/Palette Table Swap Diffuse** shader. You gonna see the inspector like the next figure depending witch shader you chose.

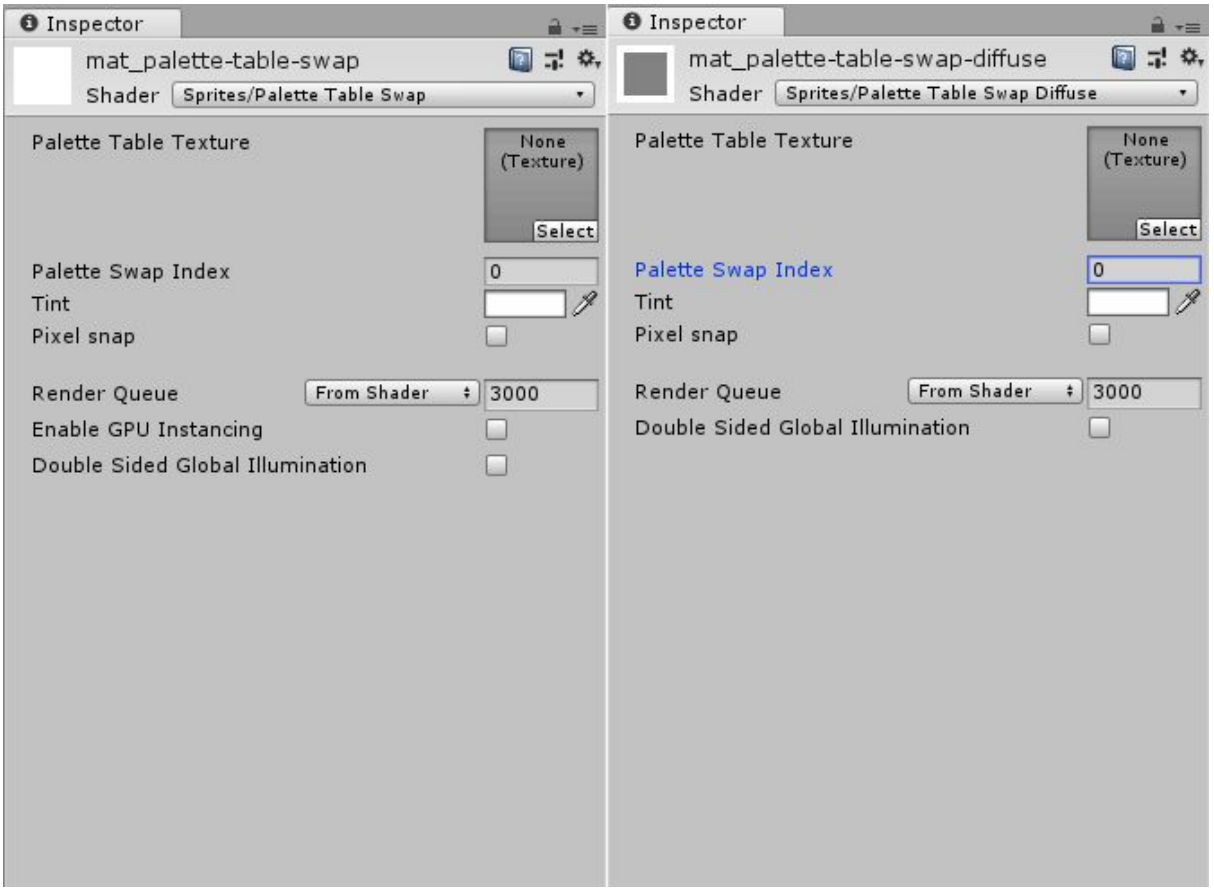

Figure 7 *- Palette Table Swap (left) and Palette Table Swap Diffuse shader inspector.*

As the previous step, you must link the material you just created with the *Sprite Renderer* component of your gameobject. If you put the table palette texture into the **Palette Table Texture** property and set the **Palette Swap Index** integer property to a number greater than 0, you should be able to see the sprite gameobject changes its colors on the scene as well. If **Palette Swap Index** property is 0 or negative, no color swap will happen.

To swap colors during gameplay, you should attach a script component called **Palette Table Swap Controller** into the gameobject with the *Sprite Renderer* component. Set the public **Palette Table** texture property from this script on the inspector and, on another script, call the public *Next Palette, Previous Palette* or *Original Palette* functions. Again, you can use the **Palette Input Swapper** script if you just want to test those features during gameplay.

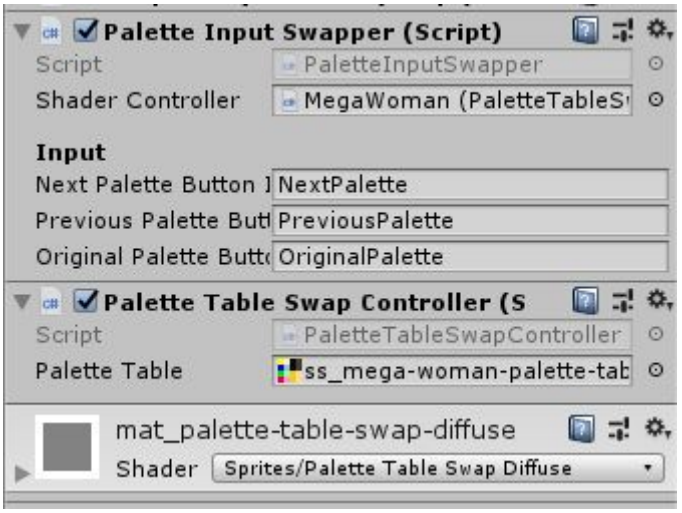

Figure 8 *- Palette Input Swapper and Palette Table Swap Controller components inspector.*

If you have any double on this process, you can find a platform player gameobject all set up in the two scenes inside the **Scene** folder. One uses a **Palette Swap Controller** while other uses the **Palette Table Swap Controller** with the diffuse shader.

How it works?

Palette colors textures should be created beforehand using some image editor. Each color should be one pixel size and the palette should contains only the colors you want to swap during gameplay.

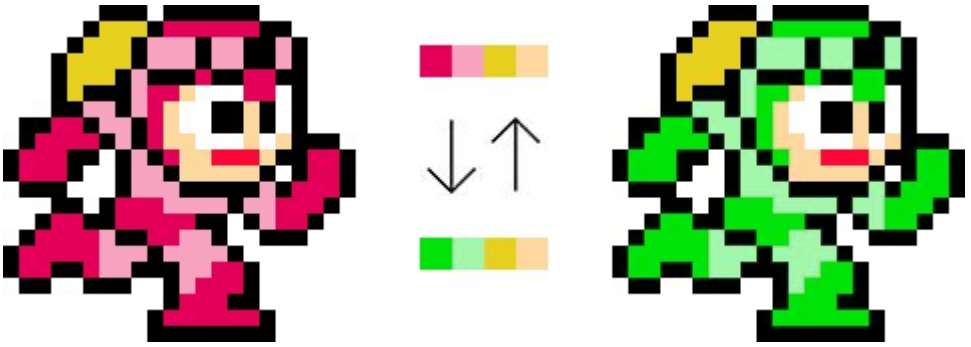

Figure 9 *- The swap color palette process.*

The shader takes the first color in the second palette and swaps to the original one. The same thing is done for the next one and so on. This process is done on shader level in the best way possible, which increase performance.

To control the swap process during gameplay, you should use **Palette Swap Controller** and **Palette Table Swap Controller** scripts components. Both extends from **Abstract Shader Swap Controller** and have the following public methods available:

Method Name	Summary
Next Palette	Swaps to the next palette.
Previous Palette	Swaps to the previous palette.
Original Palette	Swaps to the original palette.

Further Information

All prefabs are properly set up and can be found at *Assets/Prefabs* folder. Also, if you need further information in how the demo project was created, please open the **Sample Scene** or **Sample Scene - Light** available on *Assets/Scenes* folder.

You can play the demo here.

Thanks for reading!