

EB GUIDE tutorial

Rendering gamma correctly

Version 6.9.0.200120181101

Copyright © 2019 Elektrobit Automotive GmbH

Legal notice

Confidential and proprietary information

ALL RIGHTS RESERVED. No part of this publication may be copied in any form, by photocopy, microfilm, retrieval system, or by any other means now known or hereafter invented without the prior written permission of Elektrobit Automotive GmbH.

All brand names, trademarks and registered trademarks are property of their rightful owners and are used only for description.

1. Tutorial: Rendering gamma correctly

NOTE**Default window layout**

All instructions and screenshots use the default window layout. If you want to follow the instructions, we recommend to set the EB GUIDE Studio window to default layout by selecting **Layout > Reset to default layout**.

In EB GUIDE Studio you can perform gamma correction for the following:

- ▶ Display
- ▶ Image
- ▶ Texture

The following instructions show you how to configure gamma correction. For best results, work through the steps in order presented.

Approximate duration: 15 minutes.



Configuring gamma encoding for displays

The following steps guide you through the process of setting up EB GUIDE Studio to output gamma-encoded values to the display.

Prerequisite:

- The content area displays the **Main** state machine.
- The **Main** state machine contains an initial state and a view state.
- The initial state has a transition to the view state.

Step 1

Click .

The project center opens.

Step 2

In the navigation area, click **Configure > Profiles**.

Step 3

For each profile that is sRGB-aware configure the following:

Step 3.1

In the content area, click the **Scenes** tab.

Step 3.2

From the `colorMode` drop-down list box, select the following:

- ▶ If your rendering pipeline uses the hardware sRGB support of the GPU, select `32-bit sRGB (4)`
- ▶ If your rendering hardware does not support sRGB, select `32-bit sRGB (Emulated) (5)`.

To apply changes in the edit mode, restart EB GUIDE Studio.

NOTE**Rendering hardware**

The OpenGL 3 renderer always has hardware sRGB support. The OpenGL renderer, which uses the OpenGL ES 2.0 API, uses only hardware sRGB support if the hardware supports it through the appropriate OpenGL ES extensions. This is automatically detected. In case your OpenGL ES 2.0 hardware does not support sRGB, the renderer automatically falls back to `32-bit sRGB (Emulated) (5)`, which uses fragment shaders for the conversion. Note that hardware sRGB mode does not work on all systems, even if they support OpenGL ES 3.0. In that case, switch to `32-bit sRGB (Emulated) (5)`.

**Configuring gamma encoding for images**

Prerequisite:

- You completed the previous instruction.
- An image file is available in the resource folder.

Step 1

Go to the project editor and double-click the view. Drag an image from the **Toolbox** into the view.

Step 2

In the **Properties** component, from the `image` drop-down box, select an image file.

The image probably appears too bright.

This is caused by the sRGB color mode that was configured in the scene properties and now applies gamma-correction to an already gamma-corrected image.

Step 3

To configure the image to be gamma encoded, in the **Properties** component select `sRGB`.

The image is now displayed and processed correctly in blending operations.

**Configuring gamma encoding for textures**

Prerequisite:

- You completed the previous instructions.

- A 3D file, for example an `.fbx` file, with at least one textured 3D object is available with a diffuse texture.

Step 1

Drag a scene graph from the **Toolbox** into the view.

Step 2

In the **Properties** component, click **Import file** and select a 3D file.

Step 3

In the **Navigation** component locate the imported scene graph. Within the scene graph structure, select a material widget that uses a diffuse texture.

Step 4

In the **Properties** component, in the **Widget feature properties**, click **Add/Remove**.

Step 5

From the **3D** category, select **Diffuse texture**.

Step 6

Select the `diffuseSRGB` property.

The texture is treated as gamma-encoded image and is linearized before it is being used in lighting computations.