



## OpenGL ES Emulator (GX819) **Errata Notice**

This document contains all errata known at the date of issue in current release (1.3.0) of the OpenGL ES Emulator.

### **Proprietary Notice**

Words and logos marked with ® or ™ are registered trademarks or trademarks of ARM Limited in the EU and other countries, except as otherwise stated below in this proprietary notice. Other brands and names mentioned herein may be the trademarks of their respective owners.

Neither the whole nor any part of the information contained in, or the product described in, this document may be adapted or reproduced in any material form except with the prior written permission of the copyright holder.

The product described in this document is subject to continuous developments and improvements. All particulars of the product and its use contained in this document are given by ARM Limited in good faith. However, all warranties implied or expressed, including but not limited to implied warranties of merchantability, or fitness for purpose, are excluded.

This document is intended only to assist the reader in the use of the product. ARM Limited shall not be liable for any loss or damage arising from the use of any information in this document, or any error or omission in such information, or any incorrect use of the product.

### **Document Confidentiality Status**

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by ARM and the party that ARM delivered this document to.

## **ARM Web Address**

The ARM website is located at the following address: <http://www.arm.com>

## Feedback

ARM welcomes feedback on this product and its documentation.

### ***Feedback on this product***

If you have any comments or suggestions about this product, contact your supplier and give the following:

- The product name.
- The product revision or version.
- An explanation with as much information as you can provide. Include symptoms if appropriate.

### ***Feedback on this document***

If you have any comments on or about this document, please send email to [errata@arm.com](mailto:errata@arm.com) giving the following:

- The document title.
- The document number.
- The page number(s) to which your comments refer.
- A concise explanation of your comments.

General suggestion for additions and improvements are also welcome.

## CONTENTS

<b>1</b>	<b>Introduction</b>	<b>5</b>
<b>2</b>	<b>Errata Summary Table</b>	<b>6</b>
<b>3</b>	<b>Errata – Category 1</b>	<b>7</b>
<b>4</b>	<b>Errata – Category 2</b>	<b>8</b>
<b>5</b>	<b>Errata – Category 3</b>	<b>9</b>
2:	<code>eglQuerySurface</code> returns the original size of a resized native window	9
7:	<code>WGL_PBUFFER_LOST_ARB</code> pbuffer memory lost not checked	10
8:	Some EGL Attributes are not supported for <code>eglChooseConfig</code> and <code>eglGetConfigAttrib</code>	11
9:	<code>eglCreatePbufferSurface</code> fails when <code>attrib_list</code> is NULL or <code>EGL_NONE</code>	12
11:	Use of back buffer is always enabled and cannot be disabled	13
13:	<code>eglChooseConfig</code> may not return a config when <code>EGL_CONFIG_ID</code> is used	14
14:	Line numbers reported by the shader compiler do not match original source	15
15:	<code>eglTerminate</code> cleans up resources associated with display	16
16:	Some EGL Surface Attributes are not supported for <code>eglQuerySurface</code>	17
17:	<code>EGL_LARGEST_PBUFFER</code> not triggered when less graphics memory is available	18
19:	<code>eglCreatePbufferSurface</code> doesn't accept some texture attributes	19
43:	<code>eglMakeCurrent</code> succeeds with incompatible surface and context	20
49:	List of EGL configurations returned by <code>eglChooseConfig</code> is not sorted	21
54:	FBO's cannot be shared between contexts	22
114:	<code>EGL_MATCH_NATIVE_PIXMAP</code> attribute not supported	24
220:	<code>GL_ES</code> macro cannot be used in shaders when running on top of Mesa	25
318:	Rendering on Mesa with Compiz enabled causes incorrect transparencies	26
335:	Switching between GLES1 and GLES2 contexts doesn't work on some NVIDIA Cards	27

# 1 INTRODUCTION

## Scope

This document describes errata categorized by level of severity. Each description includes:

- a unique defect tracking identifier
- the current status of the defect
- where the implementation deviates from the specification and the conditions under which erroneous behavior occurs
- the implications of the erratum with respect to typical applications
- the application and limitations of a 'work-around' where possible

## Categorization of Errata

Errata recorded in this document are split into three levels of severity:

Category 1 Behavior that is impossible to work around and that severely restricts the use of the product in all, or the majority of applications, rendering the device unusable.

Category 2 Behavior that contravenes the specified behavior and that might limit or severely impair the intended use of specified features, but does not render the product unusable in all or the majority of applications.

Category 3 Behavior that was not the originally intended behavior but should not cause any problems in applications.

## 2 ERRATA SUMMARY TABLE

The errata associated with this product affect product versions as below.

A cell shown thus **X** indicates that the defect affects the platform shown at the top of that column.

<b>ID</b>	<b>Cat</b>	<b>Summary of erratum</b>	<b>Windows</b>	<b>Linux</b>
2	3	<code>eglQuerySurface</code> returns the original size of a resized native window	X	X
7	3	<code>WGL_PBUFFER_LOST_ARB</code> pbuffer memory lost not checked	X	
8	3	Some EGL Attributes are not supported for <code>eglChooseConfig</code> and <code>eglGetConfigAttrib</code>		X
9	3	<code>eglCreatePbufferSurface</code> fails when <code>attrib_list</code> is NULL or <code>EGL_NONE</code>	X	X
11	3	Use of back buffer is always enabled and cannot be disabled	X	X
13	3	<code>eglChooseConfig</code> may not return a config when <code>EGL_CONFIG_ID</code> is used		X
14	3	Line numbers reported by the shader compiler do not match original source	X	X
15	3	<code>eglTerminate</code> cleans up resources associated with display	X	
16	3	Some EGL Surface Attributes are not supported for <code>eglQuerySurface</code>		X
17	3	<code>EGL_LARGEST_PBUFFER</code> not triggered when less graphics memory is available	X	
19	3	<code>eglCreatePbufferSurface</code> doesn't accept some texture attributes		X
43	3	<code>eglMakeCurrent</code> succeeds with incompatible surface and context		X
49	3	List of EGL configurations returned by <code>eglChooseConfig</code> is not sorted		X
54	3	FBO's cannot be shared between contexts	X	X
68	3	Incorrect error code returned instead of <code>EGL_BAD_MATCH</code>		X
114	3	<code>EGL_MATCH_NATIVE_PIXMAP</code> attribute not supported	X	X
220	3	<code>GL_ES</code> macro cannot be used in shaders when running on top of Mesa	X	
318	3	Rendering on Mesa with Compiz enabled causes incorrect transparencies		X
335	3	Switching between GLES1 and GLES2 contexts doesn't work on some NVIDIA Cards	X	

## **3 ERRATA – CATEGORY 1**

**There are no Errata in this Category**

## **4 ERRATA – CATEGORY 2**

**There are no Errata in this Category**

## 5 ERRATA – CATEGORY 3

### 2: `eglQuerySurface` returns the original size of a resized native window

#### **Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0

#### **Description**

The values of `EGL_WIDTH` and `EGL_HEIGHT`, as returned by `eglQuerySurface`, are determined at the time `eglCreateWindowSurface` is called, and are not updated if the window is later resized.

#### **Implications**

Applications that wish to respond to window resize events cannot use EGL to determine the new window size.

#### **Workaround**

If window resizes are not expected or if a non-resizable window is used, no action is required. Applications that wish to respond to changes in window size must use the native windowing API to determine the new size of the window.

**7: WGL\_PBUFFER\_LOST\_ARB pbuffer memory lost not checked****Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Windows

**Description**

Pbuffers and pixmaps are supported via the use of the `WGL_ARB_pbuffer` extension (see [http://www.opengl.org/registry/specs/ARB/wgl\\_pbuffer.txt](http://www.opengl.org/registry/specs/ARB/wgl_pbuffer.txt)). This specifies that a `WGL_PBUFFER_LOST_ARB` query can be made to check for loss of memory due to a display mode change. The OpenGL ES Emulator does not query `WGL_PBUFFER_LOST_ARB`.

**Implications**

The OpenGL ES Emulator will not detect pbuffer memory lost due to a display mode change.

**Workaround**

Do not change display modes while running the OpenGL ES Emulator.

## 8: Some EGL Attributes are not supported for `eglChooseConfig` and `eglGetConfigAttrib`

### **Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Linux

### **Description**

When any of the following attributes are given as input to `eglChooseConfig` or `eglGetConfigAttrib`, `EGL_BAD_ATTRIBUTE` is set. This error is returned because the underlying GLX implementation doesn't have counterpart GLX attributes that can be used.

- `EGL_LUMINANCE_SIZE`
- `EGL_ALPHA_MASK_SIZE`
- `EGL_BIND_TO_TEXTURE_RGB`
- `EGL_BIND_TO_TEXTURE_RGBA`
- `EGL_COLOR_BUFFER_TYPE`
- `EGL_MAX_SWAP_INTERVAL`
- `EGL_MIN_SWAP_INTERVAL`

### **Implications**

The above EGL calls will set the error `EGL_BAD_ATTRIBUTE` instead of returning with `EGL_SUCCESS`.

### **Workaround**

There is no workaround available for this issue as the EGL implementation relies on the underlying GLX implementation.

**9: `eglCreatePbufferSurface` fails when `attrib_list` is `NULL` or `EGL_NONE`****Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Windows

**Description**

When `eglCreatePbufferSurface` is called with attribute list that is `NULL` or `EGL_NONE`, no surface is returned. In this scenario, all attributes assume their default values. This includes width and height for the surface which are set to 0. The underlying `WGL` implementation doesn't return a valid surface when width and height of the surface are 0.

**Implications**

Creation of Pbuffer surfaces with default attributes will not work.

**Workaround**

The width, height and other attributes of the desired Pbuffer surface should be mentioned for `eglCreatePbufferSurface` to work.

## 11: Use of back buffer is always enabled and cannot be disabled

### **Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0

### **Description**

`eglChooseConfig` translates the EGL attribute list into an attribute list suitable for the underlying OpenGL graphics driver. In addition to this, it adds another attribute to the list. This additional attribute asks the underlying OpenGL graphics driver to return only those configurations that allow use of the back buffer. Hence use of back buffer is always enabled and applications cannot disable it.

### **Implications**

Use of the back buffer is always enabled and cannot be disabled.

### **Workaround**

None.

### 13: eglChooseConfig may not return a config when EGL\_CONFIG\_ID is used

#### **Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0

#### **Description**

EGL\_CONFIG\_ID's are in the range 1 to n (number of configurations). Our implementation maps EGL\_CONFIG\_ID to GLX\_FBCONFIG\_ID's which do not share the same range (e.g. may not start from 1, the range depends on the GPU driver).

Therefore, when using `eglChooseConfig` with the attribute `EGL_CONFIG_ID` with what should be a valid `EGL_CONFIG_ID` (e.g. 1) it is possible that no configurations will be returned.

#### **Implications**

User cannot rely on choosing configurations using `EGL_CONFIG_ID` without first checking that the ID is valid.

#### **Workaround**

Use `eglChooseConfig` or `eglGetConfigs` to get configurations and then use `eglGetConfigAttribs` with the `EGL_CONFIG_ID` to get the `EGL_CONFIG_ID`'s of the configurations. Using these `EGL_CONFIG_ID` values with `eglChooseConfig` will successfully return configurations.

## 14: Line numbers reported by the shader compiler do not match original source

### **Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0

### **Description**

Due to translation of shader language from ESSL to GLSL for use by the underlying OpenGL graphics driver and the concatenation of strings input to `glShaderSource`, error line numbers may not match the original source code.

### **Implications**

Shader compiler errors reported by the emulator cannot be directly mapped back to original shader source code line numbers.

### **Workaround**

Ensure that the Mali GPU Offline Shader Compiler is installed and configured correctly. Shader source code is sent unmodified to the Offline Shader Compiler and hence line numbering should be correct. Be aware that separate strings passed into `glShaderSource` are concatenated before being passed to the Offline Shader Compiler.

**15: eglTerminate cleans up resources associated with display****Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Windows

**Description**

When `eglTerminate` is called with a valid display, all resources associated with the display are removed. As a result, the following scenarios produce an error.

- 1) Calling `eglTerminate` again results in `EGL_BAD_DISPLAY` being set instead of `EGL_TRUE`
- 2) Calling `eglInitialize` after `eglTerminate` results in `EGL_BAD_DISPLAY` being set instead of `EGL_SUCCESS`
- 3) Calling `eglQueryString` after `eglTerminate` results in `EGL_BAD_DISPLAY` being set instead of `EGL_NOT_INITIALIZED`

**Implications**

The implications for the scenarios mentioned above are,

- 1) `eglGetError` will return the wrong error when `eglTerminate` is called repeatedly
- 2) `eglInitialize` will not succeed after `eglTerminate`
- 3) `eglGetError` will return the wrong error after `eglQueryString`

**Workaround**

The resources for the display are created again when `eglGetDisplay` is called after `eglTerminate` and all the above scenarios will produce the expected output.

## 16: Some EGL Surface Attributes are not supported for eglQuerySurface

### **Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Linux

### **Description**

When any of the following attributes are given as input to `eglQuerySurface`, `EGL_BAD_ATTRIBUTE` is set. This error is returned because the underlying GLX implementation doesn't have counterpart GLX attributes that can be used.

- `EGL_LUMINANCE_SIZE`
- `EGL_ALPHA_MASK_SIZE`
- `EGL_BIND_TO_TEXTURE_RGB`
- `EGL_BIND_TO_TEXTURE_RGBA`
- `EGL_COLOR_BUFFER_TYPE`
- `EGL_MAX_SWAP_INTERVAL`
- `EGL_MIN_SWAP_INTERVAL`

### **Implications**

The above EGL calls will set the error `EGL_BAD_ATTRIBUTE` instead of returning with `EGL_SUCCESS`.

### **Workaround**

There is no workaround available for this issue as the EGL implementation relies on the underlying GLX implementation.

**17: EGL\_LARGEST\_PBUFFER not triggered when less graphics memory is available****Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Windows

**Description**

When `eglCreatePbufferSurface` is called with `EGL_LARGEST_PBUFFER` attribute set to `EGL_TRUE`, no surface is returned when the graphics memory available is less. The function is supposed to return a Pbuffer surface with width and height that would fit in the memory available.

**Implications**

Creation of Pbuffer surfaces that rely on `EGL_LARGEST_PBUFFER` will not work.

**Workaround**

The work around is to call `eglCreatePbufferSurface` repeatedly (with decreased height and width) until it succeeds.

## 19: `eglCreatePbufferSurface` doesn't accept some texture attributes

### **Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Linux

### **Description**

When `eglCreatePbufferSurface` is called with the following attributes, no surface is returned.

Linux:

- `EGL_VG_COLORSPACE`
- `EGL_VG_ALPHA_FORMAT`

The underlying `GLX` implementations don't have corresponding attributes and as a result no surface is returned.

### **Implications**

Creation of Pbuffer surfaces which contain the above attributes will not work.

### **Workaround**

There is no workaround available as the EGL implementation relies on the underlying `GLX` implementations for Pbuffer support.

**43: eglMakeCurrent succeeds with incompatible surface and context****Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Linux

**Description**

It has been observed on some platforms that `eglMakeCurrent` will succeed even if the draw or read surface is not compatible with the current context. This is due to the GLX layer failing to detect the incompatibility on these platforms.

**Implications**

On some platforms, applications cannot rely upon `eglMakeCurrent` to detect incompatibility between surface and context.

**Workaround**

None.

**49: List of EGL configurations returned by `eglChooseConfig` is not sorted****Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Linux

**Description**

The list of configurations returned by `eglChooseConfig` is not sorted.

**Implications**

Applications must not rely upon sorting of configurations by `eglChooseConfig`.

**Workaround**

None.

**54: FBO's cannot be shared between contexts****Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0

**Description**

There is no support in the emulator for sharing of FBO's between contexts although the GL ES 2.0 specification allows this.

**Implications**

The user cannot share FBO's between contexts when using the emulator.

**Workaround**

None.

**68: Incorrect error code returned instead of `EGL_BAD_MATCH`****Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Linux

**Description**

Sometimes, instead of `EGL_BAD_MATCH`, EGL returns an incorrect error code. This happens because the Linux version of EGL is implemented on top of GLX, which does not have an error code corresponding to `EGL_BAD_MATCH` and EGL is not always able to detect the real cause of the error.

**Implications**

Applications will sometimes see an incorrect error code instead of `EGL_BAD_MATCH`.

**Workaround**

None.

**114: EGL\_MATCH\_NATIVE\_PIXMAP attribute not supported****Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Windows

**Description**

The attribute `EGL_MATCH_NATIVE_PIXMAP` is not supported by `eglChooseConfig`.

**Implications**

The EGL 1.3 specification says that the attribute `EGL_MATCH_NATIVE_PIXMAP` was introduced to make it easier to choose an `EGLConfig` to match a native pixmap. This attribute is accepted by the emulator, but is ignored other than to validate the provided handle.

**Workaround**

Applications should work as expected even if the chosen `EGLConfig` does not match the pixmap format because rendering is done to an internal buffer and then copied to the pixmap, including any necessary pixel format conversions. If an 8-bit per channel `EGLConfig` is desired (to ensure the same color precision as the native pixmap), then `EGL_RED_SIZE`, `EGL_GREEN_SIZE` and `EGL_BLUE_SIZE` should be explicitly passed to `eglChooseConfig`.

**220: GL\_ES macro cannot be used in shaders when running on top of Mesa****Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0

**Description**

When using Mesa as the native OpenGL implementation the compiler will not define the GL\_ES macro.

**Implications**

If a shader contains conditionally compiled code depending on the presence of, or value of, the GL\_ES macro then this code will not be compiled. Code that should be disabled by the definition of GL\_ES will be compiled. Shaders will typically only contain such code if the same shader sources are used for multiple graphics APIs (e.g. OpenGL and OpenGL ES), and these shaders require different code for different APIs.

**Workaround**

Replace the conditional compilation with separate sources.

### **318: Rendering on Mesa with Compiz enabled causes incorrect transparencies**

#### ***Status***

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Linux

#### ***Description***

When using Linux with Mesa as the OpenGL renderer and using Compiz desktop effects, the transparencies of the scene rendered with the emulator will be incorrect. The rendered scene will be semi-transparent with any underlying windows/desktop visible through the scene.

#### ***Implications***

Using this particular configuration will not give the correct visuals.

#### ***Workaround***

Disabling Compiz desktop effects fixes the issue (i.e. setting the System->Preferences->Appearance->Visual Effects to None).

### **335: Switching between GLES1 and GLES2 contexts doesn't work on some NVIDIA Cards**

#### **Status**

Affects: OpenGL ES Emulator

Fault Status: Cat 3, Present in version 1.3.0 for Windows

#### **Description**

When using OpenGL ES 1.1 and OpenGL ES 2.0 contexts in the same application you might get incorrect rendering on some NVIDIA cards. We have only seen this effect on NVIDIA Quadro 2000 graphics card.

#### **Implications**

Using this particular configuration will not give the correct visuals.

#### **Workaround**

Use a different card or Mesa software rendering.