

EMMA Mobile™ BSP for Android™ A3D OpenGL ES1.1/2.0 Interface Specification

Mobile Application Processor
EMMA Mobile EV



Document Number: R21UT00083EJ0140

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Technology Corp. website (<http://www.renesas.com>).

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
 - "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
 - "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
 - "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

Trademarks

- EMMA Mobile is a trademark or registered trademark of Renesas Electronics Corporation in Japan, Germany, United States of America, and other countries.
- Linux® is a registered trademark or a trademark of Linus Torvalds in the United States and/or other countries.
- Android™ is a trademark of Google Inc. Use of this trademark is subject to Google Permissions
- Windows and Windows Media are registered trademarks of Microsoft Corporation in the United States and other countries.
- OpenGL® and the oval logo are trademarks or registered trademarks of Silicon Graphics, Inc. in the United States and/or other countries worldwide.
- Other company names and product names mentioned herein are registered trademarks or trademarks of their respective owners.
- Registered trademark and trademark symbols (® and ™) are omitted in this document.

How to Use This Manual

[Readers]

This manual is intended for engineers who develop products which use the mobile application processor.

[Purpose]

This manual is intended to give users an understanding of the functions of the mobile application processor device driver or OMF and to serve as a reference for developing hardware and software for systems that use this driver.

[How to Read This Manual]

It is assumed that the readers of this manual have general knowledge in the fields of electrical engineering, logic circuits, microcontrollers, and Linux.

To understand the overall functions of this driver or OMF.

→ Read this manual in the order of the **CONTENTS**.

To understand the functions of a multimedia processor for mobile applications

→ See the **Multimedia Processor for Mobile Applications User's Manual**.

To know the electrical specifications of the multimedia processor for mobile applications

→ See the **Multimedia Processor for Mobile Applications Data Sheet**.

[Conventions]

The following symbols are used in this manual.

Data significance : Higher digits on the left and lower digits on the right

Note : Footnote for item marked with **Note** in the text

Caution : Information requiring particular attention

Remark : Supplementary information

Numeric representation : Binary ... xxxx, 0bxxxx, or xxxxB

Decimal ... xxxx

Hexadecimal ... 0xxxxx or xxxxxH

Data type : Word ... 32 bits

Halfword ... 16 bits

Byte ... 8 bits

- CONTENTS -

| | |
|--|-----------|
| 1. OVERVIEW | 7 |
| 1.1. OVERVIEW | 7 |
| 1.2. FUNCTION | 7 |
| 1.3. MEMORY SIZE CHECK FUNCTION | 7 |
| 1.4. OPERATION ENVIRONMENT | 7 |
| 1.5. RESTRICTION | 8 |
| 1.5.1. OPENGLES1.1/2.0 RESTRICTION | 8 |
| 1.5.2. EGL RESTRICTION | 11 |
| 1.5.3. OTHER RESTRICTION | 11 |
| 1.6. REFERENCE | 12 |
| 1.6.1. SPECIFICATION | 12 |
| 1.6.2. RELATED INFORMATION | 13 |
| 1.7. GLOSSARY | 13 |
| 2. COMPILE PROCEDURE | 14 |
| 2.1. THE CONFIGURATION OF RELEASED PACKAGE | 14 |
| 2.2. COMPILE PROCEDURE | 15 |
| 3. FUNCTION | 16 |
| 3.1. MODULE CONFIGURATION | 16 |
| 3.2. DRAWING INTERFACE | 17 |
| 3.2.1. OPENGLES1.1 (OES) | 17 |
| 3.2.2. OPENGLES2.0 | 17 |
| 3.2.3. EGL | 17 |
| 3.3. EXTENSION FUNCTION | 18 |
| 3.3.1. OPENGLES1.1 (OES) EXTENSION FUNCTION | 18 |
| 3.3.2. OPENGLES2.0 EXTENSION FUNCTION | 20 |
| 3.4. THE DEPENDENCY ITEMS FOR IMPLEMENTATION | 23 |
| 3.4.1. THE DEPENDENCY ITEMS FOR OPENGLES1.1 IMPLEMENTATION | 23 |
| 3.4.2. THE DEPENDENCY ITEMS FOR OPENGLES2.0 IMPLEMENTATION | 27 |
| 3.4.3. THE DEPENDENCY ITEMS FOR EGL IMPLEMENTATION | 30 |
| 3.4.4. MEMORY SIZE CHECK FUNCTION | 33 |
| 4. CONTROL INTERFACE | 34 |
| 4.1. SUMMARY | 34 |
| 4.2. THE ENVIRONMENT FOR USING CONTROL INTERFACE | 34 |
| 4.3. PBUFFER DRAWABLE CONTROL INTERFACE | 34 |
| 4.3.1. SUMMARY | 34 |
| 4.3.2. CREATE DRAWABLE | 34 |
| 4.3.3. PBUFFER DRAWABLE RENDERING PROCESS | 35 |
| 4.3.4. DESTROY DRAWABLE | 35 |
| 4.4. WINDOWS DRAWABLE CONTROL INTERFACE | 36 |
| 4.4.1. SUMMARY | 36 |
| 4.4.2. CREATE DRAWABLE | 36 |
| 4.4.3. WINDOW DRAWABLE RENDERING PROCESS | 44 |
| 4.4.4. DESTROY DRAWABLE | 45 |
| 4.5. STRIDE CONTROLLING | 46 |
| 5. OTHERS | 48 |
| 5.1. HEADERFILE | 48 |
| 5.2. SHADER COMPILER | 48 |
| 6. PARAMETER LISTS | 49 |
| 6.1. EGL CONFIG LISTS | 49 |

- CONTENTS OF FIGURE -

| | |
|---|----|
| Figure 2-1 The Configuration of Released Package..... | 14 |
| Figure 3-1 Module Configuration..... | 16 |
| Figure 4-1 240x345 Pixel Image | 46 |
| Figure 4-2 60x100 Pixel Image | 47 |

- CONTENTS OF TABLE -

| | |
|---|----|
| Table 1-1 Operation Environment | 7 |
| Table 1-2 OpenGL ES1.1 Supporting Extension List (1)..... | 8 |
| Table 1-3 OpenGL ES1.1 Supporting Extension List (2)..... | 9 |
| Table 1-4 OpenGL ES2.0 Supporting Extension List | 9 |
| Table 1-5 OpenGL ES1.1/2.0 Restriction | 10 |
| Table 1-6 EGL Unsupported Items..... | 11 |
| Table 1-7 EGL Restriction Items | 11 |
| Table 1-8 Specification Information..... | 12 |
| Table 1-9 Related Information..... | 13 |
| Table 1-10 Glossary | 13 |
| Table 3-1 The OpenGL ES1.1 Supporting Extension Function List (1)..... | 18 |
| Table 3-2 The OpenGL ES1.1 Supporting Extension Function List (2)..... | 19 |
| Table 3-3 The OpenGL ES2.0 Supporting Extension Function List (1)..... | 20 |
| Table 3-4 The OpenGL ES2.0 Supporting Extension Function List (2)..... | 22 |
| Table 3-5 The Dependency Items for OpenGL ES1.1 Implementation | 23 |
| Table 3-6 Setting for OpenGL ES1.1 Implementation..... | 25 |
| Table 3-7 The Dependency Items for OpenGL ES2.0 Implementation | 27 |
| Table 3-8 Setting for OpenGL ES2.0 Implementation..... | 29 |
| Table 3-9 The Dependency Items for EGL Implementation | 30 |
| Table 3-10 eglGetProcAddress Supporting Extension Function (OpenGL ES1.1) | 30 |
| Table 3-11 eglGetProcAddress Supporting Extension Function (OpenGL ES2.0) | 33 |
| Table 4-1 NativeWindowType Structure..... | 40 |
| Table 4-2 android_native_base_t Structure | 40 |
| Table 4-3 android_native_buffer_t Structure..... | 40 |
| Table 6-1 Config ID = 0..... | 49 |
| Table 6-2 Config ID = 1..... | 50 |
| Table 6-3 Config ID = 2..... | 51 |
| Table 6-4 Config ID = 3..... | 52 |
| Table 6-5 Config ID = 4..... | 53 |
| Table 6-6 Config ID = 5..... | 54 |
| Table 6-7 Config ID = 6..... | 55 |
| Table 6-8 Config ID = 7..... | 56 |
| Table 6-9 Config ID = 8 | 57 |
| Table 6-10 Config ID = 9 | 58 |
| Table 6-11 Config ID = 10..... | 59 |
| Table 6-12 Config ID = 11..... | 60 |
| Table 6-13 Config ID = 12 | 61 |
| Table 6-14 Config ID = 13 | 62 |
| Table 6-15 Config ID = 14 | 63 |
| Table 6-16 Config ID = 15 | 64 |

1. OVERVIEW

1.1. Overview

This manual describes the 3D graphics rendering function, supported by the 3D graphics IP core SGX530 ("SGX") which is mounted on EMMA Mobile EV2 on Android platform.

1.2. Function

This system uses Open GL ES 1.1/2.0 and EGL 1.4 for off-screen 3D rendering and on-screen 3D rendering. OpenGL ES 1.1/2.0 is the approved graphic application interface based on Khronos (<http://www.khronos.org>). OpenGLES1.1 is based on OpenGL version 1.5 and OpenGLES2.0 is based on OpenGL version 2.0.

The actual OpenGLES1.1 library version used in this system is "Common Profile" version.

1.3. Memory Size Check Function

The working memory area for this system is assigned from kernel memory, and the maximum assigned working memory size is designed to assignable function, and this function is supplied as an extendible function. For details, please refer to chapter 3.4.4.

1.4. Operation Environment

The Table 1-1 lists the operation environment in this system.

Table 1-1 Operation Environment

| | |
|-----------------|---|
| Device | EMMA Mobile EV2 |
| Platform | EMMA Mobile EV BSP(Android) Android 2.2 |
| Graphic IP Core | SGX 530 |

1.5. Restriction

The following are the restriction of this system.

1.5.1. OpenGL ES1.1/2.0 Restriction

The OpenGL ES1.1 and OpenGL ES2.0 restriction are as follow.

● OpenGL ES1.1/2.0 Extension Function

About OpenGL ES1.1, this system ONLY supports extension listed in Table 1-2. About OpenGL ES2.0, this system ONLY supports extension listed in Table 1-4.

Other extensions are not supported, except these are listed in Table 1-2 and Table 1-4.

Table 1-2 OpenGL ES1.1 Supporting Extension List (1)

| Extension Type | Extension Name |
|---|----------------------------------|
| OES Ratified (required for Conformance) | OES_byte_coordinates |
| | OES_fixed_point |
| | OES_single_precision |
| | OES_matrix_get |
| | OES_read_format |
| | OES_compressed_paletted_texture |
| | OES_point_sprite |
| OES Ratified | OES_point_size_array |
| | OES_matrix_palette |
| | OES_draw_texture |
| | OES_query_matrix |
| | OES_texture_env_crossbar |
| | OES_texture_mirrored_repeat |
| | OES_texture_cube_map |
| | OES_blend_subtract |
| | OES_blend_func_separate |
| | OES_blend_equation_separate |
| | OES_stencil_wrap |
| | OES_extended_matrix_palette |
| | OES_framebuffer_object |
| | OES_rgb8_rgba8 |
| | OES_depth24 |
| | OES_stencil8 |
| | OES_compressed_ETC1_RGB8_texture |
| | OES_mapbuffer |

Table 1-3 OpenGL ES1.1 Supporting Extension List (2)

| Extension Type | Extension Name |
|----------------|-------------------------------|
| Other | IMG_read_format |
| | IMG_texture_compression_pvrtc |
| | IMG_texture_format_BGRA8888, |
| | EXT_texture_format_BGRA8888, |

Table 1-4 OpenGL ES2.0 Supporting Extension List

| Extension Type | Extension Name |
|----------------|----------------------------------|
| OES Ratified | OES_fbo_render_mipmap |
| | OES_rgb8_rgba8 |
| | OES_depth24 |
| | OES_vertex_half_float |
| | OES_texture_float |
| | OES_element_index_uint |
| | OES_mapbuffer |
| | OES_fragment_precision_high |
| | OES_compressed_ETC1_RGB8_texture |
| | OES_get_program_binary |
| | OES_packed_depth_stencil |
| | OES_depth_texture |
| | OES_standard_derivatives |
| Other | IMG_texture_compression_pvrtc |
| | IMG_read_format |
| | IMG_program_binary |
| | IMG_shader_binary |
| | IMG_texture_format_BGRA8888 |
| | EXT_texture_format_BGRA8888 |

● OpenGL ES1.1/2.0 restriction

Table 1-5 lists the OpenGL ES1.1/2.0 restriction.

Table 1-5 OpenGL ES1.1/2.0 Restriction

| Function | Content |
|--------------------------------------|---|
| glEnable glDisable | The following parameters for multi-sample are invalid. GL_SAMPLE_ALPHA_TO_COVERAGE GL_SAMPLE_ALPHA_TO_ONE GL_SAMPLE_COVERAGE |
| glSampleCoverage glSampleCoverage | Setting is invalid. |

1.5.2. EGL Restriction

The EGL restriction is shown as follows.

- **EGL unsupported items**

EGL unsupported items are shown in table 1-6.

Table 1-6 EGL Unsupported Items

| Function | Content |
|---------------|---|
| eglWaitNative | Not support. There is no guarantee when call this function |

- **EGL restriction**

The EGL restriction is shown in Table 1-7.

Table 1-7 EGL Restriction Items

| Restriction Parameter | Restriction Content |
|-----------------------|--|
| Rendering Surface | Support WindowSurface, PbufferSurface. Not support Pixmap Surface. |
| Render format | Support RGB565 and RGBA8888ONLY. Not support RGBA4444 and RGBA5551. |
| Config ID | Support 0, 1, 2, 3, 8, 9, 10, 11ONLY. Not support 4~7,12~15. |

1.5.3. Other Restriction

For other items, there is no restriction

- **Not supported items**

None.

- **Restriction items**

None.

- **The dependency items for the implementation**

None.

1.6. Reference

1.6.1. Specification

Table 1-8 lists the related specifications in this module.

Table 1-8 Specification Information

| Specification No. & Title |
|--|
| OpenGL ES Common/Common-Lite Profile Specification Version 1.1.12 (Full Specification) http://www.khronos.org/registry/gles/specs/1.1/es_full_spec_1.1.12.pdf |
| OpenGL ES Common/Common-Lite Profile Specification Version 1.1.12 (Difference Specification) http://www.khronos.org/registry/gles/specs/1.1/es_cm_spec_1.1.12.pdf |
| OpenGL ES 1.1 Online Manual Pages http://www.khronos.org/opengles/sdk/1.1/docs/man/ |
| The OpenGL Graphics System: A Specification (Version 1.5) http://www.opengl.org/documentation/specs/version1.5/glspec15.pdf |
| OpenGL ES 2.0.24 Full Specification http://www.khronos.org/registry/gles/specs/2.0/es_full_spec_2.0.24.pdf |
| OpenGL ES Difference Specification Version 2.0.24 http://www.khronos.org/registry/gles/specs/2.0/es_cm_spec_2.0.24.pdf |
| OpenGL ES Shading Language Specification Version 1.0.17 http://www.khronos.org/registry/gles/specs/2.0/GLSL_ES_Specification_1.0.17.pdf |
| OpenGL 2.0 Specification (Version 2.0) http://www.opengl.org/documentation/specs/version2.0/glspec20.pdf |
| EGL 1.4 Specification http://www.khronos.org/registry/egl/specs/eglspec.1.4.20090908.pdf |

1.6.2. Related Information

Table 1-9 lists the related information in this module.

Table 1-9 Related Information

| Related Information & Title |
|---|
| GLSurfaceView class description (According to Android Developer) http://developer.android.com/intl/ja/reference/android/opengl/GLSurfaceView.html |

1.7. Glossary

Table 1-10 lists the glossary used in this system.

Table 1-10 Glossary

| Glossary | Explanation |
|-----------------|--|
| SGX | The Official name of Graphics IP core in EM-EV2 is SGX530. |
| Drawable | Drawing area of this system. The drawing memory is confirmed by up-module. |
| pixmap surface | OpenGL ES is the drawing area to process imaginary off-screen, and it can be accessed from Native API. OpenGL ES can draw pictures on drawable according to <code>eglCreatePixmapSurface()</code> . □ This system does not support pixmap surface. |
| pbuffer surface | OpenGL ES is the drawing area to process imaginary off-screen. OpenGL ES can draw pictures on drawable according to <code>eglCreatePbufferSurface()</code> . |
| window surface | OpenGL ES is the drawing area to process imaginary on-screen OpenGL ES can draw pictures on drawable according to <code>eglCreateWindowSurface()</code> . |
| Shader compiler | Compiler of shader-program. In OpenGL ES2.0, shader can be used as programmable shader. To implement programmable shader, it is necessary to generate shader-code for a processor from shader-source program by using shader-compiler There are two ways to compile a source file . One is online-compile compiled when program is processing, and another is offline-compile compiled when program is building. |

2. COMPILE PROCEDURE

2.1. The Configuration of Released Package

Figure 2-1 lists the configuration of the released package “SGX patch mydroid_sgx”.

```
-- device
  -- renesas
    -- emev
      -- sgx
        |-- Android.mk
        |-- egl.cfg
        |-- libIMGegl.so
        |-- libEGL_SGX530.so
        |-- libGLESv1_CM_SGX530.so
        |-- libGLESv2_SGX530.so
        |-- libglslcompiler.so
        |-- libogles_frexp.so
        |-- libpvr2d.so
        |-- libpvrPVR2D_ANDROIDWSEGL.so
        |-- libsrv_init.so
        |-- libsrv_um.so
        |-- libusc.so
        |-- powervr.ini
        |-- pvrsvinit
        -- pvrsvkm.ko
```

Figure 2-1 The Configuration of Released Package

2.2. Compile Procedure

The following describes the build procedure of released package.

● Released package

EV BSP_Android_SGX_YYYYMMDD.tar.gz

| | |
|------|-------|
| YYYY | Year |
| MM | Month |
| DD | Date |

(2010/06/06 release→EV BSP_Android_SGX_20100606.tar.gz)

● Compile Step

1) Decompress the released package SGX patch, and then apply the patch files to Android root directory "mydroid"

```
# tar xzf EV BSP_Android_SGX_YYYYMMDD.tar.gz
# cp -a mydroid_sgx/* mydroid/
```

2) Run make

```
# cd mydroid
# make
```

The released package is applied by the above procedure.

3. FUNCTION

3.1. Module Configuration

Figure 3-1 shows the configuration of this module.

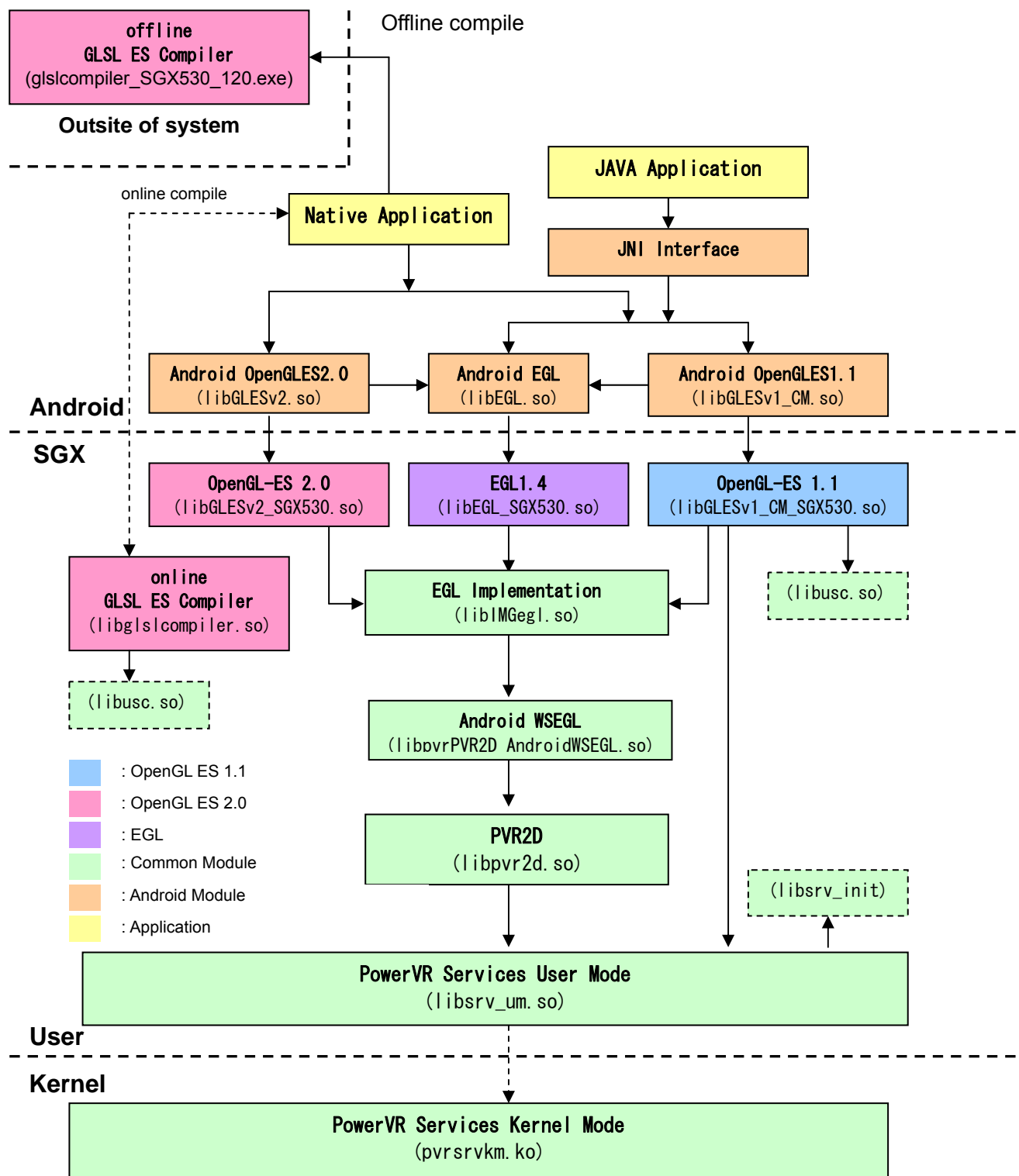


Figure 3-1 Module Configuration

3.2. Drawing Interface

3.2.1. OpenGL ES1.1 (OES)

This system supports OpenGL ES 1.1. For details, please refer to the following material supplied by khronos.

OpenGL ES Common/Common-Lite Profile Specification

OpenGL ES 1.1 & EGL Reference Manual

<http://www.khronos.org/opengles/spec/>

3.2.2. OpenGL ES2.0

This system supports OpenGL ES 2.0. For details, please refer to the following material supplied by khronos.

OpenGL ES Difference Specification Version 2.0.21

<http://www.khronos.org/opengles/spec/>

3.2.3. EGL

This system supports EGL1.4 in OpenGL ES1.1.

This system supports pbuffer surface and window surface.

For supporting EGL config, please refer to "6.1 EGL Config list".

For details about EGL, please refer to the following material supplied by khronos.

OpenGL ES Common/Common-Lite Profile Specification

OpenGL ES 1.1 & EGL Reference Manual

<http://www.khronos.org/opengles/spec/>

EGL 1.4 Specification

<http://www.khronos.org/registry/egl/specs/eglspec.1.4.20090908.pdf>

3.3. Extension Function

The following describes the supported extension functions in this system.

3.3.1. OpenGL ES1.1 (OES) Extension Function

This system has functions listed in Table 3-1 & Table.3-2. as the OpenGL ES 1.1 extension function. Marked “○” in “Support” means “supported”, marked “x” means “unsupported”. For details, please refer to the following URL:

<http://www.khronos.org/registry/gles/>

Table 3-1 The OpenGL ES1.1 Supporting Extension Function List (1)

| Extension Type | Extension Name | Supplementary | Support |
|---|---------------------------------|---|---------|
| OES ratified (required-for Conformance) | OES_byte_coordinates | Make it possible to use byte data on vertex array & texture coordinate. | ○ |
| | OES_fixed_point | Make it possible to use fixed decimal-point data on variable & vertex array. | ○ |
| | OES_single_precision | Make it possible to use single-precision floating point variable on variable. | ○ |
| | OES_matrix_get | Current-matrix can be got.(Can got Current-matrix) | ○ |
| | OES_read_format | implement a format to read-back the frame buffer. | ○ |
| | OES_compressed_paletted_texture | Support palette texture function | ○ |
| | OES_point_sprite | The purpose of this extension is to allow such applications to use points rather than quads. | ○ |
| OES Ratified | OES_point_size_array | Make it possible to receive every point's point size effectively | ○ |
| | OES_matrix_palette | Make vertex blending (skinning) possible. | ○ |
| | OES_draw_texture | Make rectangle drawing based on texture possible. | ○ |
| | OES_query_matrix | Make it possible to get element/state parameters of current matrix. | ○ |
| | OES_texture_env_crossbar | Put other texture unit as source and supply functions with colour mixture. This extension adds the ability to use the color from any texture unit as a source. | ○ |
| | OES_texture_mirrored_repeat | Add mirrored-repeat texture mode. | ○ |
| | OES_texture_cube_map | Make it possible to use mapping in cube. | ○ |
| | OES_blend_subtract | Add two kinds of blend types. | ○ |
| | OES_blend_func_separate | Blend RGB and α separately and add separate blend function. | ○ |
| | OES_blend_equation_separate | This extension provides a separate blending equation for RGB and alpha to match the generality available for blend factors. | ○ |
| | OES_stencil_wrap | Add two kinds of StencilWrapping process. | ○ |
| | OES_extended_matrix_palette | Expend the size of OES_matrix_palette array and the minimum setting of vertex array weigh. | ○ |

Table 3-2 The OpenGL ES1.1 Supporting Extension Function List (2)

| Extension Type | Extension Name | Supplementary | Support |
|----------------|----------------------------------|--|---------|
| OES Ratified | OES_framebuffer_object | Supply frame buffer function | ○ |
| | OES_rgb8_rgba8 | Make rendering of RGB8 & RGBA8 buffer possible. | ○ |
| | OES_depth24 | Make rendering of frame buffer owning 24bit depth possible. | ○ |
| | OES_stencil8 | Make rendering of frame buffer owning 8bit stencil possible | ○ |
| | OES_compressed_ETC1_RGB8_texture | Make it possible to use ETC format's compressed data. | ○ |
| | OES_mapbuffer | Make it possible to use map/unmap function of vertex buffer object. | ○ |
| | OES_EGL_image | Supply generation function of EGLImage object. | × |
| | OES_required_internalformat | Make it possible to choose required internal format. | × |
| | OES_vertex_array_object | Allow to use vertex array object | × |
| | OES_EGL_sync | Insert fence to EGL, make it possible to check if the process is over in GL command pipe line. | × |
| Other | EXT_multi_draw_arrays | Allow calling one GL function to render more than one primitive | × |
| | IMG_read_format | Add format/type combinations to glReadPixels. | ○ |
| | IMG_texture_compression_pvrtc | Allow to use PVRTC compressed data. | ○ |
| | IMG_texture_stream | Allow to use hardware source including Image data as texture. | × |
| | IMG_texture_format_BGRA8888, | Allow to use BGRA format texture. | ○ |
| | EXT_texture_format_BGRA8888 | | ○ |

3.3.2. OpenGL ES 2.0 Extension Function

This system has functions listed in Table 3-3 & Table.3-4 as the OpenGL ES 2.0 extension function. Marked “○” in “Support” means “supported”, marked “×” means “unsupported”. For details, please refer to the following URL:

<http://www.khronos.org/registry/gles/>

Table 3-3 The OpenGL ES2.0 Supporting Extension Function List (1)

| Extension Type | Extension Name | Supplementary | Support |
|----------------|----------------------------------|---|---------|
| OES Ratified | OES_fbo_render_mipmap | Allow implementations to support rendering to any mip-level of a texture(s) that is attached to a framebuffer object(s). | ○ |
| | OES_rgb8_rgba8 | This extension enables RGB8 and RGBA8 renderbuffer storage formats | ○ |
| | OES_depth24 | This extension enables 24-bit depth components as a valid render buffer storage format. | ○ |
| | OES_vertex_half_float | This extension adds a 16-bit floating pt data type (aka half float) to vertex data specified using vertex arrays. | ○ |
| | OES_texture_float | These extensions add texture formats with 16- (aka half float) and 32-bit floating-point components. | ○ |
| | OES_element_index_uint | OpenGL ES 1.0 supports DrawElements with <type> value of UNSIGNED_BYTE and UNSIGNED_SHORT. This extension adds support for UNSIGNED_INT <type> values. | ○ |
| | OES_mapbuffer | This extension adds to the vertex buffer object functionality supported by OpenGL ES 1.1 or ES 2.0 by allowing the entire data storage of a buffer object to be mapped into the client's address space. | ○ |
| | OES_fragment_precision_high | Allow to use high-precision fragment shader | ○ |
| | OES_compressed_ETC1_RGB8_texture | The goal of this extension is to allow direct support of compressed textures in the Ericsson Texture Compression (ETC) formats in OpenGL ES. | ○ |
| | OES_EGL_image | This extension provides a mechanism for creating texture and renderbuffer objects sharing storage with specified EGLImage objects (such objects are referred to as "EGLImage targets"). | × |
| | OES_required_internalformat | Make it possible to choose required internal format. | × |
| | OES_vertex_array_object | Make it possible to use vertex array | × |
| | OES_EGL_sync | Insert fence into EGL, make it possible to check if the process is over in GL command pipe line | × |
| | OES_get_program_binary | Make it possible to pick-up the data from program binary compiled. | ○ |

| | | | |
|--|--------------------------|---|---|
| | OES_packed_depth_stencil | Allow to use internal format which is packed by the depth and stencil. | ○ |
| | OES_depth_texture | This extension defines a new texture format that stores depth values in the texture. | ○ |
| | OES_standard_derivatives | The standard derivative built-in functions and semantics from OpenGL 2.0 are optional for OpenGL ES 2.0. When this extension is available, these built-in functions are also available, as is a hint controlling the quality/performance trade off. | ○ |

Table 3-4 The OpenGL ES2.0 Supporting Extension Function List (2)

| Extension Type | Extension Name | Supplementary | Support |
|----------------|--|---|---------|
| Other | IMG_program_binary | This extension makes available a program binary format, SGX_PROGRAM_BINARY_IMG. It enables retrieving and loading of pre-linked program objects on chips designed by Imagination Technologies. | ○ |
| | IMG_shader_binary | This extension enables loading precompiled binary shaders compatible with chips designed by Imagination Technologies. | ○ |
| | IMG_texture_compression_pvrtc | This extension provides additional texture compression functionality specific to Imagination Technologies PowerVR Texture compression format (called PVRTC) subject to all the requirements and limitations described by the OpenGL 1.3 specifications. | ○ |
| | IMG_texture_stream2 | The texture data can be streamed from buffer class device to GL. | × |
| | IMG_read_format | This extension is intended to supplement the GL_OES_read_format extension by adding support for more format/type combinations to be used when calling ReadPixels. | ○ |
| | IMG_texture_npot | Allow to use texture whose mipmap filter is not 2 powers. | × |
| | IMG_texture_format_BGRA8888 EXT_texture_format_BGRA8888 | Allow to use BGRA format texture. | ○ |
| | EXT_multi_draw_arrays | Their main purpose is to allow one function call to render more than one primitive such as triangle strip, triangle fan, etc. | × |
| | EXT_discard_framebuffer | Make it possible to process the Frame buffer attachment after it has been used. | × |

3.4. The Dependency Items for Implementation

The dependency items for EGL and OpenGL ES 1.1 implementation are described as following.

3.4.1. The Dependency Items for OpenGL ES1.1 Implementation

The dependency items for OpenGL ES1.1 implementation is shown in Table 3-5. In addition, the details setting for implementation are shown in Table 3-6.

Table 3-5 The Dependency Items for OpenGL ES1.1 Implementation

| Function | Content |
|---|--|
| glActiveTexture | Support 4 texture units |
| glClientActiveTexture | Support 4 texture units |
| glClipPlanef glClipPlanex glGetClipPlane | Support 6 Clip Planes |
| glCompressedTexImage2D glCompressedTexSubImage2D | <p>The maximum setting size is 2048x2048</p> <p>The following 15 formats can be used</p> <ul style="list-style-type: none"> •GL_PALETTE4_RGB8_OES •GL_PALETTE4_RGBA8_OES •GL_PALETTE4_R5_G6_B5_OES •GL_PALETTE4_RGBA4_OES •GL_PALETTE4_RGB5_A1_OES •GL_PALETTE8_RGB8_OES •GL_PALETTE8_RGBA8_OES •GL_PALETTE8_R5_G6_B5_OES •GL_PALETTE8_RGBA4_OES •GL_PALETTE8_RGB5_A1_OES •GL_ETC1_RGB8_OES •GL_COMPRESSED_RGB_PVRTC_2BPPV1_IMG •GL_COMPRESSED_RGBA_PVRTC_2BPPV1_IMG •GL_COMPRESSED_RGB_PVRTC_4BPPV1_IMG •GL_COMPRESSED_RGBA_PVRTC_4BPPV1_IMG |
| glCopyTexImage2D | The maximum setting size is 2048x2048. |
| glCopyTexSubImage2D | The maximum setting size is 2048x2048. |
| glDrawArrays glDrawElements | <p>The maximum index is 65536.</p> <p>The maximum vertex array is 65536.</p> |
| glLightf , glLightx glLightfv , glLightxv | Support 8 light. |
| glLineWidth glLineWidthx | <p>The valid size is 1~16.</p> <p>The valid Anti-Alias Line size is 1.</p> |
| glMultiTexCoord4f glMultiTexCoord4v | GL_ES1_MAX_TEXTURE_UNITS is 4. |
| glPointSize glPointSizex | <p>The valid size is 1~32.</p> <p>The valid Anti-Alias Line size is 1.</p> |
| glPopMatrix glPushMatrix | <p>The capacity of stack size is as following</p> <ul style="list-style-type: none"> •MAX_MODELVIEW_STACK_DEPTH=16 •MAX_PROJECTION_STACK_DEPTH=2 •MAX_TEXTURE_STACK_DEPTH=4 |

| Function | Content |
|---|--|
| glReadPixels | Parameter type can be set to following values •GL_UNSIGNED_BYTE (for RGBA8888 surface) •GL_UNSIGNED_SHORT_4_4_4_4 (for RGBA4444 surface) •GL_UNSIGNED_SHORT_5_6_5 (for RGB565 surface) Parameter format can be set to following values •GL_RGBA (for RGBA8888 surface) •GL_RGBA (for RGBA4444 surface) •GL_RGB (for RGB565 surface) |
| glStencilOp glStencilFunc glStencilMask glClearStencil | Support 0, 8bit for stencil buffer. |
| glTexImage2D glTexSubImage2D | The maximum size is 2048×2048. |
| glViewport | The maximum size is 2048×2048 |
| CurrentPaletteMatrixOES | The maximum PaletteMatrix size is 32. The index value can be set to less than 32. |

Table 3-6 Setting for OpenGL ES1.1 Implementation

| Parameter | Value |
|--------------------------------|-------------------------------------|
| SUBPIXEL_BITS | 4 |
| MAX_TEXTURE_SIZE | 2048 |
| MAX_CUBE_MAP_TEXTURE_SIZE | 2048 |
| MAX_VIEWPORT_DIMS | 2048 |
| ALIASED_POINT_SIZE_RANGE | 1 – 32 |
| ALIASED_LINE_WIDTH_RANGE | 1 – 16 |
| SMOOTH_POINT_SIZE_RANGE | 1 |
| SMOOTH_LINE_SIZE_RANGE | 1 |
| MAX_ELEMENTS_INDICES | 65536 |
| MAX_ELEMENTS_VERTICES | 65536 |
| SAMPLE_BUFFERS | 0 – 1 |
| SAMPLES | 0 – 4 (2x2 anti aliasing) |
| NUM_COMPRESSED_TEXTURE_FORMATS | 15 |
| COMPRESSED_TEXTURE_FORMATS | GL_PALETTE4_RGB8_OES |
| | GL_PALETTE4_RGBA8_OES |
| | GL_PALETTE4_R5_G6_B5_OES |
| | GL_PALETTE4_RGBA4_OES |
| | GL_PALETTE4_RGB5_A1_OES |
| | GL_PALETTE8_RGB8_OES |
| | GL_PALETTE8_RGBA8_OES |
| | GL_PALETTE8_R5_G6_B5_OES |
| | GL_PALETTE8_RGBA4_OES |
| | GL_PALETTE8_RGB5_A1_OES |
| | GL_COMPRESSED_RGB_PVRTC_2BPPV1_IMG |
| | GL_COMPRESSED_RGBA_PVRTC_2BPPV1_IMG |
| | GL_COMPRESSED_RGB_PVRTC_4BPPV1_IMG |
| | GL_COMPRESSED_RGBA_PVRTC_4BPPV1_IMG |
| | GL_ETC1_RGB8_OES |
| GLES1_MAX_TEXTURE_UNITS | 4 |
| MAX_CLIP_PLANES | 6 |
| MAX_LIGHTS | 8 |
| MAX_MODELVIEW_STACK_DEPTH | 16 |
| MAX_PROJECTION_STACK_DEPTH | 2 |
| MAX_TEXTURE_STACK_DEPTH | 4 |
| MAX_PALETTE_MATRICES_OES | 32 |
| MAX_VERTEX_UNITS_OES | 4 |
| STENCIL_BITS | 0, 8 |
| RENDERER | PowerVR SGX 530 |
| VENDOR | Imagination Technologies |
| VERSION | OpenGL ES-CM 1.1 |
| | OpenGL ES-CL 1.1 |

| Parameter | Value |
|--|--|
| IMPLEMENTATION_COLOR_READ_TYPE_OES | GL_UNSIGNED_BYTE (for RGBA8888surface) |
| | GL_UNSIGNED_SHORT_4_4_4_4 (for RGBA4444surface) |
| | GL_UNSIGNED_SHORT_5_6_5 (for RGB565surface) |
| IMPLEMENTATION_COLOR_READ_FORMAT_OES | GL_RGBA (for RGBA8888surface) |
| | GL_RGBA (for RGBA4444surface) |
| | GL_RGB (for RGB565surface) |
| RED_BITS, GREEN_BITS, BLUE_BITS, ALPHA_BITS | 8,8,8,8 |
| | 4,4,4,4 |
| | 5,5,5,1 |
| | 5,6,5,0 |
| DEPTH_BITS | 0, 16, 24 |

3.4.2. The Dependency Items for OpenGL ES2.0 Implementation

The dependency items for OpenGL ES2.0 implementation is shown in Table 3-7, In addition, the detail setting for implementation are shown in Table 3-8.

Table 3-7 The Dependency Items for OpenGL ES2.0 Implementation

| Function | Content |
|---|--|
| glActiveTexture | Support 8 texture units |
| glClientActiveTexture | Support 8 texture units |
| glCompressedTexImage2D glCompressedTexSubImage2D | The maximum setting size is 2048x2048. The following formats can be used <ul style="list-style-type: none"> •GL_ETC1_RGB8_OES •GL_COMPRESSED_RGB_PVRTC_2BPPV1_IMG •GL_COMPRESSED_RGBA_PVRTC_2BPPV1_IMG •GL_COMPRESSED_RGB_PVRTC_4BPPV1_IMG •GL_COMPRESSED_RGBA_PVRTC_4BPPV1_IMG |
| glCopyTexImage2D | The maximum setting size is 2048x2048. |
| glCopyTexSubImage2D | The maximum setting size is 2048x2048. |
| glDrawArrays glDrawElements | The maximum index value is 65536. The maximum vertex array value is 65536. |
| glLineWidth glLineWidthx | The valid value is 0~16 (Anti-Alias Line is invalid) |
| glPointSize glPointSizex | The valid value is 0~511 (Anti-Alias Point is invalid) |
| glReadPixels | Parameter type can be set to the following value <ul style="list-style-type: none"> •GL_UNSIGNED_BYTE (for RGBA8888 surface) •GL_UNSIGNED_SHORT_4_4_4_4 (for RGBA4444 surface) •GL_UNSIGNED_SHORT_5_6_5 (for RGB565 surface) Parameter format can be set to the following value <ul style="list-style-type: none"> •GL_RGBA (for RGBA8888 surface) •GL_RGBA (for RGBA4444 surface) •GL_RGB (for RGB565 surface) |
| glStencilOp glStencilFunc glStencilMask glClearStencil | Stencil buffer supports 0, 8bits. |
| glTexImage2D glTexSubImage2D | The maximum setting size is 2048x2048. |
| glViewport | The maximum setting size is 2048x2048. |

| Function | Content |
|--|---|
| glVertexAttrib1fv glVertexAttrib2fv glVertexAttrib3fv glVertexAttrib4fv | As a parameter, the maximum setting value for vector is 8. |
| glUniform1fv glUniform2fv glUniform3fv glUniform4fv glUniform1iv glUniform2iv glUniform3iv glUniform4iv glUniformMatrix2fv glUniformMatrix3fv glUniformMatrix4fv | The maximum setting value for vertex vector is 128. The maximum setting value for Fragment vector is 64. |

Table 3-8 Setting for OpenGL ES2.0 Implementation

| Parameter | Value |
|---|---|
| SUBPIXEL_BITS | 4 |
| MAX_TEXTURE_SIZE | 2048 |
| MAX_CUBE_MAP_TEXTURE_SIZE | 2048 |
| MAX_VIEWPORT_DIMS | 2048 |
| ALIASED_POINT_SIZE_RANGE | 0 – 511 |
| ALIASED_LINE_WIDTH_RANGE | 0 – 16 |
| MAX_ELEMENTS_INDICES | 65536 |
| MAX_ELEMENTS_VERTICES | 65536 |
| SAMPLE_BUFFERS | 0 – 1 |
| SAMPLES | 0 – 4 (2x2 anti aliasing) |
| COMPRESSED_TEXTURE_FORMATS | GL_COMPRESSED_RGB_PVRTC_2BPPV1_IMG |
| | GL_COMPRESSED_RGBA_PVRTC_2BPPV1_IMG |
| | GL_COMPRESSED_RGB_PVRTC_4BPPV1_IMG |
| | GL_COMPRESSED_RGBA_PVRTC_4BPPV1_IMG |
| | GL_ETC1_RGB8_OES |
| NUM_COMPRESSED_TEXTURE_FORMATS | 5 |
| RENDERER | PowerVR SGX 530 |
| SHADING_LANGUAGE_VERSION | OpenGL ES GLSL 1.00 |
| VENDOR | Imagination Technologies |
| VERSION | OpenGL ES 2.0 |
| MAX_VERTEX_ATTRIBS | 8 |
| MAX_VERTEX_UNIFORM_VECTORS | 128 |
| MAX_VARYING_VECTORS | 8 |
| MAX_COMBINED_TEXTURE_IMAGE_UNITS | 8 |
| MAX_VERTEX_TEXTURE_IMAGE_UNITS | 8 |
| MAX_TEXTURE_IMAGE_UNITS | 8 |
| MAX_FRAGMENT_UNIFORM_VECTORS | 64 |
| RED_BITS, GREEN_BITS, BLUE_BITS, ALPHA_BITS | 8,8,8,8 |
| | 4,4,4,4 |
| | 5,5,5,1 |
| | 5,6,5,0 |
| DEPTH_BITS | 0, 16, 24 |
| STENCIL_BITS | 0, 8 |
| IMPLEMENTATION_COLOR_READ_TYPE_OES | GL_UNSIGNED_BYTE (for ARGB8888 surface) |
| | GL_UNSIGNED_SHORT_4_4_4_4 (for ARGB4444 surface) |
| | GL_UNSIGNED_SHORT_5_6_5 (for RGB565 surface) |
| IMPLEMENTATION_COLOR_READ_FORMAT_OES | GL_BGRA (for ARGB8888 surface) |
| | GL_BGRA (for ARGB4444 surface) |
| | GL_RGB (for RGB565 surface) |

3.4.3. The Dependency Items for EGL Implementation

The dependency items for EGL implementation is shown in Table 3-9. In addition, for the support extension shown in Table 3-9, the OpenGL ES1.1 extension function is shown in Table 3-10, and the OpenGL ES2.0 extension function is shown in Table 3-11.

Table 3-9 The Dependency Items for EGL Implementation

| Function | Content |
|-------------------|--|
| eglGetDisplay | Support EGL_DEFAULT_DISPLAY ONLY. |
| eglGetProcAddress | The eglGetProcAddress supporting extension functions are listed in Table 3-10 and 3-11 |

Table 3-10 eglGetProcAddress Supporting Extension Function (OpenGL ES1.1)

| Extension Name | eglGetProcAddress Support Extension Function |
|----------------------|--|
| OES_point_size_array | glPointSizePointerOES |
| OES_query_matrix | glQueryMatrixxOES |
| OES_matrix_palette | glCurrentPaletteMatrixOES |
| | glLoadPaletteFromModelViewMatrixOES |
| | glMatrixIndexPointerOES |
| | glWeightPointerOES |
| OES_draw_texture | glDrawTex{sifx}OES |
| | glDrawTex{sifx}vOES |
| | glTexParameter{ifx}v |
| OES_fixed_point | glVertex{234}x[v]OES |
| | glNormal3x[v]OES |
| | glTexCoord{1234}x[v]OES |
| | glMultiTexCoord{1234}x[v]OES |
| | glColor{34}x[v]OES |
| | glIndexx[v]OES |
| | glRectxOES |
| | glRectxvOES |
| | glDepthRangexOES |
| | glLoadMatrixxOES |
| | glMultMatrixxOES |
| | glLoadTransposeMatrixxOES |
| | glMultTransposeMatrixxOES |
| | glRotatexOES |
| | glScalexOES |
| | glTranslatexOES |
| | glFrustumxOES |
| | glOrthoxOES |
| | glTexGenx[v]OES |
| | glGetTexGenxvOES |
| | glClipPlanexOES |

| Extension Name | eglGetProcAddress Support Extension Function |
|------------------------|--|
| OES_fixed_point(go on) | glGetClipPlanexOES |
| | glRasterPos{234}x[v]OES |
| | glMaterialx[v]OES |
| | glGetMaterialxOES |
| | glLightx[v]OES |
| | glGetLightxOES |
| | glLightModelx[v]OES |
| | glPointSizeOES |
| | glPointParameterxvOES |
| | glLineWidthxOES |
| | glPolygonOffsetxOES |
| | glPixelStorex |
| | glPixelTransferxOES |
| | glPixelMapx |
| | glGetPixelMapxv |
| | glConvolutionParameterx[v]OES |
| | glGetConvolutionParameterxvOES |
| | glGetHistogramParameterxvOES |
| | glPixelZoomxOES |
| | glBitmapxOES |
| | glTexParameterx[v]OES |
| | glGetTexParameterxvOES |
| | glGetTexLevelParameterxvOES |
| | glPrioritizeTexturesxOES |
| | glTexEnvx[v]OES |
| | glGetTexEnvxvOES |
| | glFogx[v]OES |
| | glSampleCoverageOES |
| | glAlphaFuncxOES |
| | glBlendColorxOES |
| | glClearColorxOES |
| | glClearDepthxOES |
| | glClearAccumxOES |
| | glAccumxOES |
| | glMap1xOES |
| | glMap2xOES |
| | glMapGrid1xOES |
| | glMapGrid2xOES |
| | glGetMapxvOES |
| | glEvalCoord{12}x[v]OES |
| | glFeedbackBufferxOES |
| | glPassThroughxOES |

| Extension Name | eglGetProcAddress Support Extension Function | |
|-----------------------------|--|-------------|
| OES_fixed_point(go on) | glGetFixedvOES | |
| OES_single_precision | glDepthRangeOES | |
| | glFrustumfOES | |
| | glOrthofOES | |
| | glClipPlaneOES | |
| | glGetClipPlaneOES | |
| | glClearDepthfOES | |
| OES_texture_cube_map | glTexGenfOES | |
| | glTexGenfvOES | |
| | glTexGeniOES | |
| | glTexGenivOES | |
| | glTexGenxOES | |
| | glTexGenxvOES | |
| | glGetTexGenfvOES | |
| | glGetTexGenivOES | |
| | glGetTexGenxvOES | |
| OES_blend_subtract | glBlendEquationOES | |
| OES_blend_func_separate | glBlendFuncSeparateOES | |
| OES_blend_equation_separate | glBlendEquationSeparateOES | |
| OES_framebuffer_object | glIsRenderbufferOES | |
| | glBindRenderbufferOES | |
| | glDeleteRenderbuffersOES | |
| | glGenRenderbuffersOES | |
| | glRenderbufferStorageOES | |
| | glGetRenderbufferParameterivOES | |
| | glIsFramebufferOES | |
| | glBindFramebufferOES | |
| | glDeleteFramebuffersOES | |
| | glGenFramebuffersOES | |
| | glCheckFramebufferStatusOES | |
| | glFramebufferTexture2DOES | |
| | glFramebufferRenderbufferOES | |
| | glGetFramebufferAttachmentParameterivOES | |
| | glGenerateMipmapOES | |
| OES_mapbuffer | glGetBufferPointervOES | |
| | glMapBufferOES | |
| | glUnmapBufferOES | |
| OES_EGL_image | glEGLImageTargetTexture2DOES | Not support |
| | glEGLImageTargetRenderbufferStorageOES | Not support |
| EXT_multi_draw_arrays | glMultiDrawArraysEXT | Not support |
| | glMultiDrawElementsEXT | Not support |

Table 3-11 eglGetProcAddress Supporting Extension Function (OpenGL ES2.0)

| Extension Name | eglGetProcAddress Support Extension Function | |
|-------------------------|--|-------------|
| OES_mapbuffer | glGetBufferPointervOES | |
| | glMapBufferOES | |
| | glUnmapBufferOES | |
| OES_EGL_image | glEGLImageTargetTexture2DOES | Not support |
| | glEGLImageTargetRenderbufferStorageOES | Not support |
| EXT_multi_draw_arrays | glMultiDrawArraysEXT | Not support |
| | glMultiDrawElementsEXT | Not support |
| OES_get_program_binary | glGetProgramBinaryOES | |
| | glProgramBinaryOES | |
| OES_vertex_array_object | glBindVertexArrayOES | Not support |
| | glDeleteVertexArraysOES | Not support |
| | glGenVertexArraysOES | Not support |
| | glIsVertexArrayOES | Not support |
| OES_EGL_sync | eglCreateSyncKHR | Not support |
| | eglDestroySyncKHR | Not support |
| | eglClientWaitSyncKHR | Not support |
| | eglGetSyncAttribKHR | Not support |
| EXT_discard_framebuffer | glDiscardFramebufferEXT | Not support |

3.4.4. Memory Size Check Function

The work memory area for SGX is assigned from kernel memory, but this may influence the whole system. In this system, the powervr.ini records the maximum work memory size, which sets an upper limit for the work memory size that will be got. In addition, the work memory size unit in powervr.ini is Mbyte.

The following is an example for powervr.ini. The upper limit for work memory size is 20Mbyte.

[default]

WorkMemoryLimit =20

NOTE) the specified size is a round upper limit value, there is no strict for under size.

4. CONTROL INTERFACE

4.1. Summary

This chapter describes how to use the control interface.
The following describes the notes when using control interface.

4.2. The Environment for Using Control Interface

When Android is booting, the "init.emev.sh" script included in released package is executed automatically. Following this, the pvrsrvkm driver is installed and pvrsrvinit is executed, and make it possible to use the control interface which will be mentioned in next section.

```
➤ init.emev.sh (extract related part)

    :
    :
    insmod /lib/modules/pvrsrvkm.ko
    chmod 666 /dev/pvrsrvkm
    /system/bin/pvrsrvinit
```

4.3. Pbuffer Drawable Control Interface

4.3.1. Summary

This system supports off-screen rendering drawable of pbuffer type.

The application specifies the Width and Height of Pbuffer via EGL_WIDTH, EGL_HEIGHT of attribute, which will be used as a parameter of eglCreatePbufferSurface to implement off-screen rendering

4.3.2. Create Drawable

The application specifies the Width and Height of Pbuffer via EGL_WIDTH, EGL_HEIGHT of attribute, then call eglCreatePbufferSurface().

The following is an example for how to use Puffer in Native interface.

```
EGLSurface
CreateSurface(
    EGLDisplay display, EGLConfig config,
    unsigned int width, unsigned int height)
{
    EGLint pb_attribs[] = { EGL_WIDTH, width,
                           EGL_HEIGHT, height,
                           EGL_NONE};

    return eglCreatePbufferSurface(display, config, pb_attribs);
}
```

Because Pbuffer is not supported by GLSurfaceView class, the above process is necessary when using from Java, JNI interface.

4.3.3. Pbuffer Drawable Rendering Process

From upper application, the following is the process for how to use Pbuffer drawable rendering.

| No. | Function&Processing | Summary |
|-----|--------------------------------|--|
| 1 | eglGetDisplay | Get the EGL display connection(EGLDisplay) |
| 2 | eglInitialize | Initialize EGL display connection (EGLDisplay) |
| 3 | eglChooseConfig | Get the EGL Frame buffer configure(EGLConfig) |
| 4 | Generate the Attribs | Generate the Attribs(Width, Height) |
| 5 | eglCreatePbufferSurface | Create the Off-screen rendering surface (EGLSurface). (Use the Attribs generated in step 4) Supply the surface area. |
| 6 | eglCreateContext | Create EGL rendering context () |
| 7 | eglMakeCurrent | Associate a context to Target surface |
| 8 | Call gl function | Implement rendering processing via OpenGL ES |
| 9 | eglWaitGL | Wait rendering processing over. |
| 10 | eglMakeCurrent | Remove target via setting surface・context parameter to NULL |
| 11 | eglDestroyContext | Destroy EGL rendering context |
| 12 | eglDestroySurface | Destroy Off-screen rendering surface (EGLSurface) |
| 13 | eglTerminate | Terminate EGL display connection |

4.3.4. Destroy Drawable

In upper application, to remove rendering memory, `eglDestroySurface()` should be called, which will destroy the pbuffer surface created by “4.3.2 Create Drawable”.

The following is an example for how to use `eglDestroySurface` from Native interface.

```

Int
DestroySurface(
    EGLDisplay display,
    EGLSurface puffersurface)
{
    /*
    **Remove pbuffer surface
    */
    eglDestroySurface(display, puffersurface);
    :
    return 0;
}

```

Because Pbuffer is not supported by `GLSurfaceView` class, the above process is necessary when using from Java, JNI interface.

4.4. Windows Drawable Control Interface

4.4.1. Summary

This system supports drawable of window type.

Rendering area (window drawable) is created by upper level module, which bases on `android_native_window_t` specified by Android solely, and are specified by the `NativeWindowType` structure.

The upper Application specifies this structure as the parameters of `eglCreateWindowSurface` to achieve the on-screen rendering.

4.4.2. Create Drawable

The application specifies the rendering area information via the `NativeWindowType` structure, then call `eglCreateWindowSurface()`.

The following code describes the `NativeWindowType` structure, related to `android_native_base_t` and `android_native_buffer_t` structure. For details, please refer to Table 4-1, 4-2, 4-3.

mydroid/framework/base/include/ui/egl/android_natives.h (extract related part)

```
-----
typedef struct android_native_window_t
{
#ifdef __cplusplus
    android_native_window_t()
        : flags(0), minSwapInterval(0), maxSwapInterval(0), xdpi(0), ydpi(0)
    {
        common.magic = ANDROID_NATIVE_WINDOW_MAGIC;
        common.version = sizeof(android_native_window_t);
        memset(common.reserved, 0, sizeof(common.reserved));
    }
#endif

    struct android_native_base_t common;

    const uint32_t flags;
    const int minSwapInterval;
    const int maxSwapInterval;
    const float xdpi;
    const float ydpi;
    intptr_t oem[4];

    int (*setSwapInterval)(struct android_native_window_t* window, int interval);
    int (*dequeueBuffer)(struct android_native_window_t* window,
        struct android_native_buffer_t** buffer);
    int (*queueBuffer)(struct android_native_window_t* window,
        struct android_native_buffer_t* buffer);
    int (*query)(struct android_native_window_t* window, int what, int* value);
    int (*perform)(struct android_native_window_t* window, int operation, ... );

    void* reserved_proc[3];
} android_native_window_t;
```

mydroid/framework/base/include/EGL/eglplatform.h: (extract related part)

```
-----  
typedef struct android_native_window_t* EGLNativeWindowType;  
:  
typedef EGLNativeWindowType NativeWindowType;  
-----
```

mydroid/framework/base/include/ui/egl/android_natives.h (extract related part)

```
-----  
typedef struct android_native_base_t  
{  
  
    int magic;  
    int version;  
    void* reserved[4];  
  
    void (*incRef)(struct android_native_base_t* base);  
    void (*decRef)(struct android_native_base_t* base);  
} android_native_base_t;  
-----
```

mydroid/framework/base/include/ui/android_native_buffer.h (extract related part)

```
-----  
typedef struct android_native_buffer_t  
{  
#ifdef __cplusplus  
    android_native_buffer_t() {  
        common.magic = ANDROID_NATIVE_BUFFER_MAGIC;  
        common.version = sizeof(android_native_buffer_t);  
        memset(common.reserved, 0, sizeof(common.reserved));  
    }  
#endif  
  
    struct android_native_base_t common;  
  
    int width;  
    int height;  
    int stride;  
    int format;  
    int usage;  
  
    void* reserved[2];  
  
    buffer_handle_t handle;  
  
    void* reserved_proc[8];  
} android_native_buffer_t;  
-----
```

Table 4-1 NativeWindowType Structure

| Members of NativeWindowType Structure | Content |
|---|--|
| const uint32_t flags; | The flag of Surface attribute |
| const int minSwapInterval; | min Swap Interval |
| const int maxSwapInterval; | max Swap Interval |
| const float xdpi; | Horizontal Resolution information |
| const float ydpi; | Vertical resolution information |
| intptr_t oem[4]; | OEM driver reserved parameters |
| int (*setSwapInterval) (struct android_native_window_t* window, int interval); | The entry pointer of setSwapInterval function. |
| int (*dequeueBuffer) (struct android_native_window_t* window, struct android_native_buffer_t** buffer); | The entry pointer of dequeueBuffer function |
| int (*queueBuffer) (struct android_native_window_t* window, struct android_native_buffer_t* buffer); | The entry pointer of queueBuffer function |
| int (*query) (struct android_native_window_t* window, int what, int* value); | The entry pointer of query function |
| int (*perform) (struct android_native_window_t* window, int operation, ...); | The entry pointer of perform function |
| void* reserved_proc[3]; | Reserved parameters |

Table 4-2 android_native_base_t Structure

| Members of android_native_base_t Structure | Content |
|---|---|
| int magic; | Specify the magic number |
| int version; | Specify the size of android_native_buffer_t structure |
| void* reserved[4]; | Reserved parameter. Default value is 0. |
| void (*incRef) (struct android_native_base_t* base); | The entry pointer of incRef function. |
| void (*decRef) (struct android_native_base_t* base); | The entry pointer of decRef function |

Table 4-3 android_native_buffer_t Structure

| Members of android_native_buffer_t Structure | Content |
|--|--|
| int width; | The horizontal size of image |
| int height; | The vertical size of image |
| int stride; | Stride. Specify the value divided by BytesPerPixel |
| int format; | Image format |
| int usage; | Specify how to use Buffer memory |
| void* reserved[2]; | Reserved parameter. Default value is 0. |
| buffer_handle_t handle; | Buffer handle |
| void* reserved_proc[8]; | Reserved parameter. Default value is 0. |

Below is an example of how to use windows drawable from Native Interface. The EGLNativeWindowType structure should be got by calling android_createDisplaySurface function (implemented in inner Android), the structure includes rendering area information. At this time, the native window is consistent with the frame buffer. The EGLNativeWindowType will be used as the parameter of eglCreateWindowSurface and create the surface.

```
EGLSurface
CreateSurface(
    EGLDisplay display, EGLAttrib Attrs)
{
    EGLNativeWindowType window;
    EGLConfig config;

    window = android_createDisplaySurface();
    EGLUtils::selectConfigForNativeWindow(display,s_Attrbs, window, &config);
    return eglCreateWindowSurface(display, config, window, NULL);
}
```

Below is an example of how to use windows drawable from Java Interface. In Android, the rendering code is abstracted by GLSurfaceView class. In below example, the rendering thread is setup by setRenderer method, and getting Window Surface processing is processed in setRenderer method. In addition, for GLSurfaceView, please refer to the following URL

<http://developer.android.com/intl/ja/reference/android/opengl/GLSurfaceView.html>

```
import javax.microedition.khronos.opengles.GL;
import android.app.Activity;
import android.opengl.GLSurfaceView;
import android.os.Bundle;

public class SpriteTextActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        mGLSurfaceView = new GLSurfaceView(this);
        mGLSurfaceView.setGLWrapper(new GLSurfaceView.GLWrapper() {
            public GL wrap(GL gl) {
                return new SampleGL(gl);
            }
        });
        mGLSurfaceView.setRenderer(new SampleRenderer(this));
        setContentView(mGLSurfaceView);
    }
}
```

Below is an example of how to use windows drawable from JNI Interface. JNI interface is same as Java interface. In Android, the rendering code is abstracted by GLSurfaceView class. In below example, the rendering thread is setup by setRenderer method, and the getting Window Surface processing is processed in thread (GLJNIView class).

Native method is build into related library, which makes it possible to run JNI from Java application .from Java code, there are some function name labeled on keyword “native”, which will use System.loadLibrary to load the Native method library. According to this, the Native method will be called in Java code.

```
public class GLJNIActivity extends Activity {

    GLJNIView mView;

    @Override protected void onCreate(Bundle icle) {
        super.onCreate(icle);        mView = new GLJNIView(getApplication());
        setContentView(mView);
    }
    @Override protected void onPause() {
        super.onPause();        mView.onPause();
    }
    @Override protected void onResume() {
        super.onResume();        mView.onResume();
    }
}

GLJNIView extends GLSurfaceView {
    GLJNIView(Context context) {
        super(context);
        init();
    }

    public GLJNIView(Context context, AttributeSet attrs) {
        super(context, attrs);
        init();
    }

    private void init() {
        setEGLContextFactory(new ContextFactory());
        setEGLConfigChooser(new ConfigChooser());
        setRenderer(new Renderer());
    }

    private static class ContextFactory implements GLSurfaceView.EGLContextFactory {
        private static int EGL_CONTEXT_CLIENT_VERSION = 0x3098;
        public EGLContext createContext(EGL10 egl, EGLDisplay display,
                                         EGLConfig eglConfig) {
            int[] attrib_list = {EGL_CONTEXT_CLIENT_VERSION, 2, EGL10.EGL_NONE };
            EGLContext context = egl.eglCreateContext(display,
                                                         eglConfig, EGL10.EGL_NO_CONTEXT, attrib_list);
            checkEglError("After eglCreateContext", egl);
            return context;
        }
        public void destroyContext(EGL10 egl, EGLDisplay display, EGLContext context) {
            egl.eglDestroyContext(display, context);
        }
    }

    private static class ConfigChooser implements GLSurfaceView.EGLConfigChooser {
        private static int EGL_OPENGL_ES2_BIT = 4;
        private static int[] configAttribs2 =
        {
            EGL10.EGL_RENDERABLE_TYPE, EGL_OPENGL_ES2_BIT,
            EGL10.EGL_NONE
        }
    }
}
```

```
};
public EGLConfig chooseConfig(EGL10 egl, EGLDisplay display) {

    int[] num_config = new int[1];
    egl.eglChooseConfig(display, configAttribs2, null, 0, num_config);

    int numConfigs = num_config[0];

    EGLConfig[] configs = new EGLConfig[numConfigs];
    egl.eglChooseConfig(display, configAttribs2, configs, numConfigs, num_config);
    return configs[0];
}

private class Renderer implements GLSurfaceView.Renderer {
    private static final String TAG = "Renderer";
    public void onDrawFrame(GL10 gl) {
        GLJNILib.FunctionA(args);
    }

    public void onSurfaceChanged(GL10 gl, argtype args) {
        GLJNILib.FunctionB(args);
    }

    public void onSurfaceCreated(GL10 gl, EGLConfig config) {
        // Do nothing.
    }
}

@Override
public boolean onKeyDown(int keyCode, KeyEvent event) {
    GLJNILib.FunctionC();
    return true;
}
}

public class GLJNILib {

    static {
        System.loadLibrary("gljni");
    }

    public static native void FunctionA(argtype args);
    public static native void FunctionB(argtype args);
    public static native void FunctionC();
}

JNIEXPORT void JNICALL Java_com_android_gljni_GLJNILib_FunctionA
(JNIEnv * env, jobject obj, argtype args)
{
    :
    /* Native writing process */
    :
}
```

4.4.3. Window Drawable Rendering Process

From upper application, the following is the process for how to use Window drawable rendering.

In addition, when using GLSurfaceView(Java interface), 1~7 steps must be called in onCreate method, 10~13 will be called after onDetachedFromWindow method becomes trigger. Step 8 should be called in onDrawFrame method, and step 9 should be called in onDrawFrame method.

| No. | Function&Processing | Summary |
|-----|---|--|
| 1 | eglGetDisplay | Get the EGL display connection (EGLDisplay). |
| 2 | eglInitialize | Initialize EGL display connection (EGLDisplay) |
| 3 | android_createDisplaySurface | Create EGLNativeWindowType structure, the necessary parameters will be set, and native Surface area is created. |
| 4 | EGLUtils:: selectConfigForNativeWindow | Get the supporting configure of EGL Framebuffer setting(EGL Config), and choose a Config which is consistent with the PixelFormat. |
| 5 | eglCreateWindowSurface | Create the on-screen rendering surface (EGLSurface) (Using the EGLNativeWindowType created be step 3). |
| 6 | eglCreateContext | Create EGL rendering context |
| 7 | eglMakeCurrent | Associate a context to Target surface. |
| 8 | Call gl function | Implement the rendering based on OpenGL ES |
| 9 | eglSwapBuffers | Wait the target surface rendering over. |
| 10 | eglMakeCurrent | Set parameter surface · context to NULL to remove the target. |
| 11 | eglDestroyContext | Destroy EGL rendering context |
| 12 | eglDestroySurface | Destroy on screen rendering surface |
| 13 | eglTerminate | Terminate EGL display connection. |

4.4.4. Destroy Drawable

In upper application, to remove rendering memory, `eglDestroySurface()` should be called, which will Destroy the pbuffer surface created by “4.4.2 Create Drawable”.

The below is a Native interface using example.

```
Int
DestroySurface(
    EGLDisplay display,
    EGLSurface windowSurface)
{
    /*
    ** destroy window surface
    */
    eglDestroySurface(display, windowSurface);
    :
    :
    return 0;
}
```

The protected method `onDetachedFromWindow` should be triggered from Java、JNI interface, then `eglDestroySurface` should be called in GL thread.

4.5. Stride Controlling

The following describes the ByteStride controlling of rendering area.

ByteStride is the 1 line alignment defined on SGX, there is 8 pixels (16 bytes in following Fig.4-1) limit. Therefore it is necessary to confirm the above description alignment limit according to the horizontal size of the rendering area.

Figure 4-1 is an image that rendering area is 240pixel×345pixel (1 pixel correspond to 2 bytes). In this case, the width satisfies the 8 pixels alignment limit, so the rendering area is 240pixel×345pixel continuous space.

width(240 pixel) × height(345 pixel) image

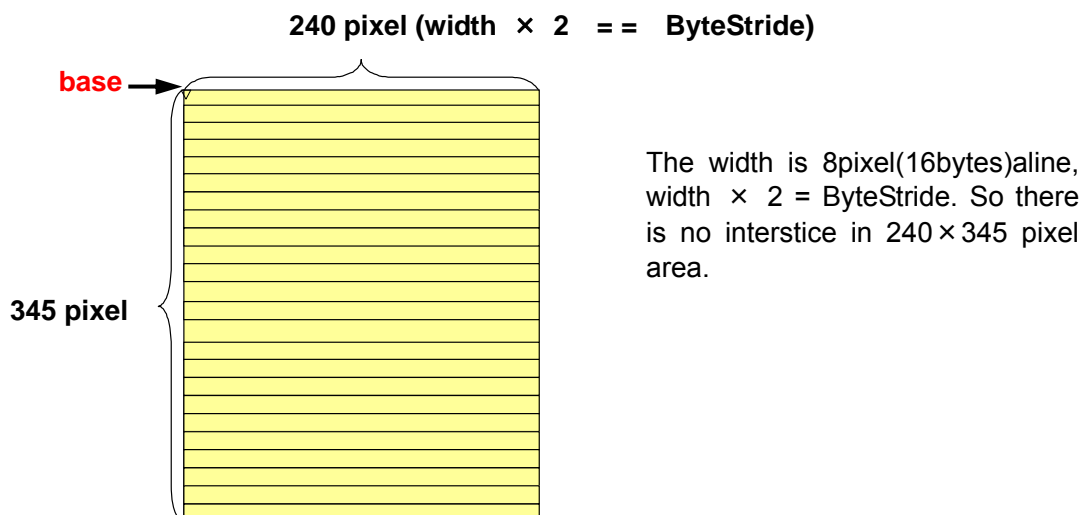


Figure 4-1 240x345 Pixel Image

In addition, Fig.4-2 is 60pixel * 100pixel (1 pixel equals to 2 bytes) of OpenGL image. In this case, since width can not meet alignment control's 8pixel unit, it is necessary to gap to meet 8pixel align. So the OpenGL turns to 60pixel * 100pixel space. But the actual drawing space is 60pixel * 100 pixel which has been removed 4Pixel from right hand.

Width (60 pixel) xHeight (100 pixel) image

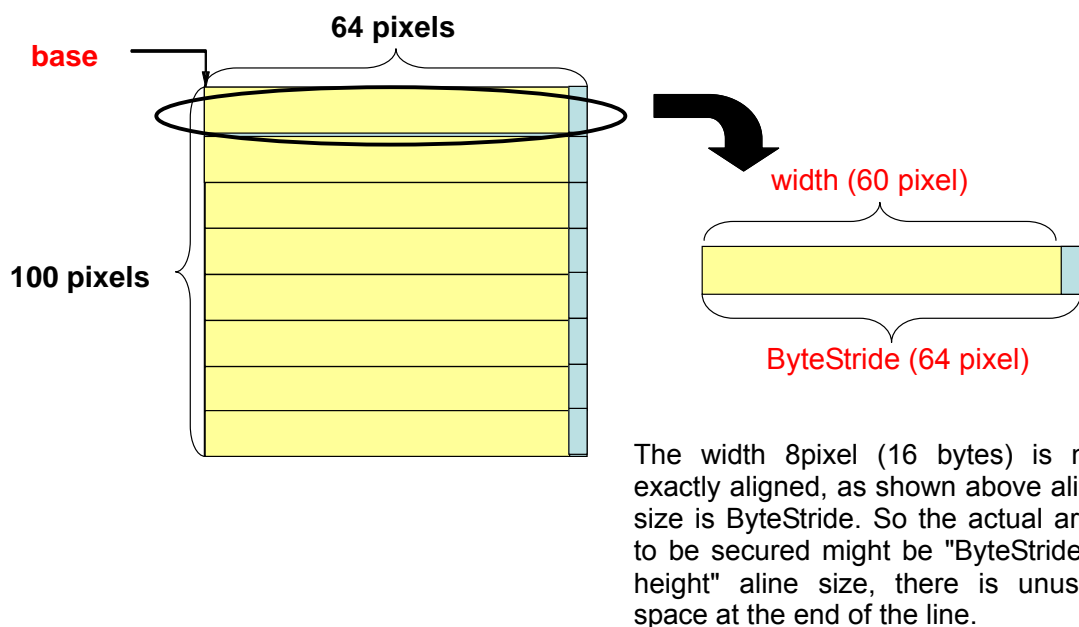


Figure 4-2 60x100 Pixel Image

5. OTHERS

5.1. HeaderFile

The structure and parameters recorded in this manual can be possibly used by including

- egl.h, eglx.h and GLES /gl.h for EGL
- GLES/gl.h and GLES/glext.h for OpenGL ES1.1
- GLES2/gl.h and GLES2/glext.h for OpenGL ES2.0

5.2. Shader Compiler

Shader Program in OpenGL2.0 is supported. The online-compile of Shader is indispensable, but Shader's online-compile in OpenGL ES2.0 is optional. The Shader program's API (glShaderBinary pixel) after login compile is prepared additionally. To compile Shader program in an offline environment, it is necessary to use Windows XP PC compile issued with OpenGL ES1.1/ES2.0 driver.

The usage of Offline compiler is as follows.

Offline compiler example:

```
$glslcompiler_SGX530_120.exe <sourcefile> <outputfile> <-v,-f>

<sourcefile> shader source code name
<outputfile> output file name

-v: specify it when compile vertex shaders
-f: specify it when compile fragment shaders
```


6. PARAMETER LISTS

6.1. EGL Config Lists

This system support Config 0, 1,2,3 8, 9,10,11 ONLY. Others are not supported.
The Table6-1~ Table 6-8 list the EGLConfig.

Table 6-1 Config ID = 0
CONFIG ID = 0 (RGB565, Depth=0)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 0 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 0 |
| EGL_RED_SIZE | 5 |
| EGL_GREEN_SIZE | 6 |
| EGL_BLUE_SIZE | 5 |
| EGL_DEPTH_SIZE | 0 |
| EGL_STENCIL_SIZE | 0 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 0 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGL_BIT |
| EGL_CONFORMANT | EGL_OPENGL_BIT |

Table 6-2 Config ID = 1
CONFIG ID = 1 (RGB565, Depth=24)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 1 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 0 |
| EGL_RED_SIZE | 5 |
| EGL_GREEN_SIZE | 6 |
| EGL_BLUE_SIZE | 5 |
| EGL_DEPTH_SIZE | 24 |
| EGL_STENCIL_SIZE | 8 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 0 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGL_BIT |
| EGL_CONFORMANT | EGL_OPENGL_BIT |

Table 6-3 Config ID = 2
CONFIG ID = 2 (ARGB8888, Depth=0)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 2 |
| EGL_BUFFER_SIZE | 32 |
| EGL_ALPHA_SIZE | 8 |
| EGL_RED_SIZE | 8 |
| EGL_GREEN_SIZE | 8 |
| EGL_BLUE_SIZE | 8 |
| EGL_DEPTH_SIZE | 0 |
| EGL_STENCIL_SIZE | 0 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGL_BIT |
| EGL_CONFORMANT | EGL_OPENGL_BIT |

Table 6-4 Config ID = 3
CONFIG ID = 3 (ARGB8888, Depth=24)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 3 |
| EGL_BUFFER_SIZE | 32 |
| EGL_ALPHA_SIZE | 8 |
| EGL_RED_SIZE | 8 |
| EGL_GREEN_SIZE | 8 |
| EGL_BLUE_SIZE | 8 |
| EGL_DEPTH_SIZE | 24 |
| EGL_STENCIL_SIZE | 8 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGL_BIT |
| EGL_CONFORMANT | EGL_OPENGL_BIT |

Table 6-5 Config ID = 4
[Unsupported] CONFIG ID = 4 (ARGB4444, Depth=0)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 4 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 4 |
| EGL_RED_SIZE | 4 |
| EGL_GREEN_SIZE | 4 |
| EGL_BLUE_SIZE | 4 |
| EGL_DEPTH_SIZE | 0 |
| EGL_STENCIL_SIZE | 0 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_CONFORMANT | EGL_OPENGL_ES_BIT |
| EGL_RENDERABLE_TYPE | EGL_OPENGL_ES_BIT |

Table 6-6 Config ID = 5
[Unsupported] CONFIG ID = 5 (ARGB4444, Depth=24)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 5 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 4 |
| EGL_RED_SIZE | 4 |
| EGL_GREEN_SIZE | 4 |
| EGL_BLUE_SIZE | 4 |
| EGL_DEPTH_SIZE | 24 |
| EGL_STENCIL_SIZE | 8 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGL_BIT |
| EGL_CONFORMANT | EGL_OPENGL_BIT |

Table 6-7 Config ID = 6
[Unsupported] CONFIG ID = 6 (ARGB1555, Depth=0)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 6 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 1 |
| EGL_RED_SIZE | 5 |
| EGL_GREEN_SIZE | 5 |
| EGL_BLUE_SIZE | 5 |
| EGL_DEPTH_SIZE | 0 |
| EGL_STENCIL_SIZE | 0 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGL_BIT |
| EGL_CONFORMANT | EGL_OPENGL_BIT |

Table 6-8 Config ID = 7
[Unsupported] CONFIG ID = 7 (ARGB1555, Depth=24)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 7 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 1 |
| EGL_RED_SIZE | 5 |
| EGL_GREEN_SIZE | 5 |
| EGL_BLUE_SIZE | 5 |
| EGL_DEPTH_SIZE | 24 |
| EGL_STENCIL_SIZE | 8 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGL_BIT |
| EGL_CONFORMANT | EGL_OPENGL_BIT |

Table 6-9 Config ID = 8
CONFIG ID = 8 (RGB565, Depth=0)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 8 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 0 |
| EGL_RED_SIZE | 5 |
| EGL_GREEN_SIZE | 6 |
| EGL_BLUE_SIZE | 5 |
| EGL_DEPTH_SIZE | 0 |
| EGL_STENCIL_SIZE | 0 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 0 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGLS2_BIT |
| EGL_CONFORMANT | EGL_OPENGLS2_BIT |

Table 6-10 Config ID = 9

CONFIG ID = 9 (RGB565, Depth=24)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 9 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 0 |
| EGL_RED_SIZE | 5 |
| EGL_GREEN_SIZE | 6 |
| EGL_BLUE_SIZE | 5 |
| EGL_DEPTH_SIZE | 24 |
| EGL_STENCIL_SIZE | 8 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 0 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGL_ES2_BIT |
| EGL_CONFORMANT | EGL_OPENGL_ES2_BIT |

Table 6-11 Config ID = 10
CONFIG ID = 10 (ARGB8888, Depth=0)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 10 |
| EGL_BUFFER_SIZE | 32 |
| EGL_ALPHA_SIZE | 8 |
| EGL_RED_SIZE | 8 |
| EGL_GREEN_SIZE | 8 |
| EGL_BLUE_SIZE | 8 |
| EGL_DEPTH_SIZE | 0 |
| EGL_STENCIL_SIZE | 0 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGLS2_BIT |
| EGL_CONFORMANT | EGL_OPENGLS2_BIT |

Table 6-12 Config ID = 11
CONFIG ID = 11 (ARGB8888, Depth=24)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 11 |
| EGL_BUFFER_SIZE | 32 |
| EGL_ALPHA_SIZE | 8 |
| EGL_RED_SIZE | 8 |
| EGL_GREEN_SIZE | 8 |
| EGL_BLUE_SIZE | 8 |
| EGL_DEPTH_SIZE | 24 |
| EGL_STENCIL_SIZE | 8 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGLS2_BIT |
| EGL_CONFORMANT | EGL_OPENGLS2_BIT |

Table 6-13 Config ID = 12
[Unsupported] CONFIG ID = 12 (ARGB4444, Depth=0)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 12 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 4 |
| EGL_RED_SIZE | 4 |
| EGL_GREEN_SIZE | 4 |
| EGL_BLUE_SIZE | 4 |
| EGL_DEPTH_SIZE | 0 |
| EGL_STENCIL_SIZE | 0 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_CONFORMANT | EGL_OPENGLS2_BIT |
| EGL_RENDERABLE_TYPE | EGL_OPENGLS2_BIT |

Table 6-14 Config ID = 13
[Unsupported] CONFIG ID = 13(ARGB4444, Depth=24)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 13 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 4 |
| EGL_RED_SIZE | 4 |
| EGL_GREEN_SIZE | 4 |
| EGL_BLUE_SIZE | 4 |
| EGL_DEPTH_SIZE | 24 |
| EGL_STENCIL_SIZE | 8 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGL2_BIT |
| EGL_CONFORMANT | EGL_OPENGL2_BIT |

Table 6-15 Config ID = 14
[Unsupported] CONFIG ID = 14 (ARGB1555, Depth=0)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 14 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 1 |
| EGL_RED_SIZE | 5 |
| EGL_GREEN_SIZE | 5 |
| EGL_BLUE_SIZE | 5 |
| EGL_DEPTH_SIZE | 0 |
| EGL_STENCIL_SIZE | 0 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGLS2_BIT |
| EGL_CONFORMANT | EGL_OPENGLS2_BIT |

Table 6-16 Config ID = 15
[Unsupported] CONFIG ID = 15 (ARGB1555, Depth=24)

| EGL_CONFIG_ATTRIBS | VALUE |
|-----------------------------|-------------------------------------|
| EGL_CONFIG_ID | 15 |
| EGL_BUFFER_SIZE | 16 |
| EGL_ALPHA_SIZE | 1 |
| EGL_RED_SIZE | 5 |
| EGL_GREEN_SIZE | 5 |
| EGL_BLUE_SIZE | 5 |
| EGL_DEPTH_SIZE | 24 |
| EGL_STENCIL_SIZE | 8 |
| EGL_LUMINANCE_SIZE | 0 |
| EGL_ALPHA_MASK_SIZE | 0 |
| EGL_CONFIG_CAVEAT | EGL_NONE |
| EGL_LEVEL | 0 |
| EGL_MAX_PBUFFER_WIDTH | 2048 |
| EGL_MAX_PBUFFER_HEIGHT | 2048 |
| EGL_MAX_PBUFFER_PIXELS | 2048*2048 |
| EGL_NATIVE_RENDERABLE | EGL_FALSE |
| EGL_NATIVE_VISUAL_ID | 0 |
| EGL_NATIVE_VISUAL_TYPE | EGL_NONE |
| EGL_SAMPLES | 0 |
| EGL_SAMPLE_BUFFERS | 0 |
| EGL_SURFACE_TYPE | EGL_WINDOW_BIT EGL_PBUFFER_BIT |
| EGL_TRANSPARENT_TYPE | EGL_NONE |
| EGL_TRANSPARENT_RED_VALUE | 0 |
| EGL_TRANSPARENT_GREEN_VALUE | 0 |
| EGL_TRANSPARENT_BLUE_VALUE | 0 |
| EGL_MIN_SWAP_INTERVAL | 1 |
| EGL_MAX_SWAP_INTERVAL | 1 |
| EGL_BIND_TO_TEXTURE_RGB | 1 |
| EGL_BIND_TO_TEXTURE_RGBA | 1 |
| EGL_COLOR_BUFFER_TYPE | EGL_RGB_BUFFER |
| EGL_MATCH_NATIVE_PIXMAP | EGL_NONE |
| EGL_RENDERABLE_TYPE | EGL_OPENGLS2_BIT |
| EGL_CONFORMANT | EGL_OPENGLS2_BIT |

| History | | EMMA Mobile BSP for Android A3D OpenGL ES1.1/2.0 Interface Specification | |
|---------|---------|---|--|
| Rev. | Date | Change Content | |
| | | Page | Point |
| - | 2010.02 | - | New for EM-EV |
| 0.90 | 2010.06 | - | <ul style="list-style-type: none"> 「NEC」 → 「renesas」 for company name and page foot Record other change point |
| 1.00 | 2010.06 | - | Change information of Android 1.6 in order to fit with Android 2.1. Add the description for OpenGL ES2.0. When Android changed from 1.6 to 2.1, the description for drawing should be changed. |
| 1.10 | 2010.07 | - | Add the description related to work memory size check. |
| | | - | Update the information of function Support Extension |
| | | - | Change the structure of release file |
| | | - | Change related Android version to 2.2 |
| 1.20 | 2010.08 | - | Renew the information of supported Extension functions. (Delete the Extensions which Android does not support) |
| | | - | Fix typos of the configuration information of CONFIG_ID=6,7,14,15. |
| 1.30 | 2010.10 | - | Support the ARGB8888 format. Support CONFIG ID=2,3,10,11. |
| 1.40 | 2010.12 | - | Fixed typos Render format in Table 1-7. RGBA8888 -> RGBA5551. [Unsupported] is not necessary in Table 6-3, Table 6-4, Table 6-11, Table 6-12. |

(1/1)

EMMA Mobile Series



Renesas Electronics Corporation

R21UT0083EJ0140