



Mali GPU Offline Shader Compiler Errata

Document Number: PR389-PRDC-011263 1.0
Date of Issue: 15th September 2009
Product: Mali GPU Offline Shader Compiler
Product Version: 2.2

© Copyright ARM Limited 2009. All rights reserved.

Abstract

This document describes the known errata in the EAC release of the Mali GPU Offline Shader Compiler version 2.2.

This is a working document throughout the product lifecycle and, as such, the content may be modified as new information is uncovered.

The information contained herein is the property of ARM Ltd. and is supplied without liability for errors or omissions. No part may be reproduced or used except as authorized by contract or other written permission. The copyright and the foregoing restriction on reproduction and use extend to all media in which this information may be embodied.

Release Information

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 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

1	Preface	4
1.1	Change Control	4
1.2	Scope	4
1.3	Terms and Abbreviations	4
2	Categorisation of errata	5
2.1	Errata Summary	5
3	CATEGORY 1 Errata	6
4	CATEGORY 2 ERRATA	7
5	CATEGORY 3 ERRATA	8
5.1	4182: Compilation of a vertex shader may fail with the error "Register allocation failed for vertex shader"	8
5.2	4484: Shader compiler preprocessor accepts some illegal combinations of preprocessor directives	8
5.3	6095: Failing to write gl_Position is reported as a shader compilation error	8
5.4	6492: Shader compiler extension macros only defined for enabled extensions	9
5.5	6564: Function overloading based on array sizes not supported	9

1 PREFACE

1.1 Change Control

<i>Issue</i>	<i>Date</i>	<i>Change</i>
1.0	15 th September 2009	Initial release.

1.2 Scope

This document describes the errata discovered in the implementation of the Mali GPU Offline Shader Compiler, categorized by level of severity. Each description includes:

- 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

1.3 Terms and Abbreviations

This document uses the following terms and abbreviations:

<i>Term</i>	<i>Meaning</i>
Cat	Category

2 CATEGORISATION OF ERRATA

Errata recorded in this document are split into three groups:

- Category 1** Features which are impossible to work around and severely restricts the use of the software in all or the majority of applications rendering the software unusable.
- Category 2** Features which contravene the specified behavior and may limit or severely impair the intended use of specified features but does not render the software unusable in all or the majority of applications.
- Category 3** Features that were not the originally intended behavior but should not cause any problems in applications.

2.1 Errata Summary

The following tables summarize all errata associated with this product.

Mali GPU Offline Shader Compiler Errata Summary

ID	Cat	Summary of Erratum
4182	Cat 3	Compilation of a vertex shader may fail with the error "Register allocation failed for vertex shader"
4484	Cat 3	Shader compiler preprocessor accepts some illegal combinations of preprocessor directives
6095	Cat 3	Failing to write gl_Position is reported as a shader compilation error
6492	Cat 3	Shader compiler extension macros only defined for enabled extensions
6564	Cat 3	Function overloading based on array sizes not supported

3 CATEGORY 1 ERRATA

No Category 1 errata

4 CATEGORY 2 ERRATA

No Category 2 errata

5 CATEGORY 3 ERRATA

5.1 4182: Compilation of a vertex shader may fail with the error "Register allocation failed for vertex shader"

Status

Affects: Mali GPU Offline Shader Compiler
Fault status: Cat 3, Present in: 2.2
Platforms Affected: Windows XP, Linux

Description

MaliGP2 has only limited internal bandwidth between its registers and execution units. In some rare cases, the register allocator for MaliGP2 in the shader compiler runs into a situation where there are more operations executed in one cycle than can be fed from the registers simultaneously. The compiler aborts the compilation in this case.

Implications

Some very big or very complex vertex shaders fail to compile.

Workaround

Changing the shader code slightly will often eliminate the problem. Try to rewrite some of the shader to do the calculations differently.

5.2 4484: Shader compiler preprocessor accepts some illegal combinations of preprocessor directives

Status

Affects: Mali GPU Offline Shader Compiler
Fault status: Cat 3, Present in: 2.2
Platforms Affected: Windows XP, Linux

Description

The ESSL compiler may in some cases accept shaders containing illegal combinations of `#if` / `#ifdef` / `#ifndef` / `#elif` / `#else` / `#endif` as valid.

Implications

This defect has no implications for valid programs.

Workaround

No workaround needed.

5.3 6095: Failing to write `gl_Position` is reported as a shader compilation error

Status

Affects: Mali GPU Offline Shader Compiler
Fault status: Cat 3, Present in: 2.2
Platforms Affected: Windows XP, Linux

Description

If the `gl_Position` built-in variable is not written anywhere in a vertex shader, the compiler reports an error. This error should only be a warning, since this (useless) situation is not explicitly forbidden by the language specification.

Implications

No implications for any useful shader code.

Workaround

No workaround needed.

5.4 6492: Shader compiler extension macros only defined for enabled extensions

Status

Affects: Mali GPU Offline Shader Compiler
Fault status: Cat 3, Present in: 2.2
Platforms Affected: Windows XP, Linux

Description

For each language extension supported by an implementation, a macro with the extension name is automatically defined by the compiler to indicate that the extension is supported. However, the compiler erroneously defines these macros only for extensions that are explicitly enabled.

Implications

Only code that is written to compile on several different implementations supporting different extensions are affected. In such code, it is not possible to explicitly test for support of an extension and conditionally compile different code depending on whether it is supported or not.

Workaround

Unconditionally enable all extensions attempted used in the shader program.

5.5 6564: Function overloading based on array sizes not supported

Status

Affects: Mali GPU Offline Shader Compiler
Fault status: Cat 3, Present in: 2.2
Platforms Affected: Windows XP, Linux

Description

The ESSL language allows functions to be overloaded based on array sizes in their parameter types, i.e. one can have two functions with identical names and identical parameter types except for the sizes of array types. The compiler treats functions with same parameter types except for array sizes as having the same signature.

Implications

Shaders containing functions with identical names and identical parameter names except for array sizes are rejected by the compiler. For instance, compilation of the program:

```
vec4 foo(float a[2])
{
    return vec4(0.0, 1.0, 0.0, 0.0);
}

vec4 foo(float a[3])
{
    return vec4(1.0, 0.0, 0.0, 0.0);
}

void main()
{
    float a[2];
    gl_Position = foo(a);
}
```

will fail with the error message

```
0:9: S0023: Function 'foo' redefined
```

Workaround

Change the function signatures to use different names.