

# 3D 图形处理器 API 符合性验证方法关键技术研究

刘 晖,田 泽,黎小玉,陈 佳

(中航工业西安航空计算技术研究所,陕西 西安 710119)

**摘 要:**3D 图形 API (Application Programming Interface, 应用程序编程接口) 是三维图形处理函数接口的简称, 它将图形系统的上层应用程序和底层驱动隔离, 使得应用软件具有良好的可移植性。OpenGL (Open Graphics Library, 开放图形接口库) 是为了规避不同硬件驱动而定义的与上层接口的标准库函数。如何开发出符合 OpenGL 标准的 API, 是 3D 图形处理器研制及应用的关键。首先系统分析了 OpenGL 标准 API 的属性, 提出了自研 3D 图形处理器 API 与 OpenGL 标准 API 符合性验证关键技术, 基于 FPGA 平台从函数的接口参数、绘图功能、显示功能和错误反馈处理这四个方来验证。上述验证方法具有很强的操作性, 验证覆盖范围更广, 验证用例的设计更为复杂。该方法加速了验证过程, 提高了验证的充分性。

**关键词:**OpenGL 核心库; 标准符合性验证; 读写可靠性; 功能完整性; 鲁棒性

**中图分类号:**TP301

**文献标识码:**A

**文章编号:**1673-629X(2014)10-0193-04

doi:10.3969/j.issn.1673-629X.2014.10.047

## Research on Key Technology of Verification Method for 3D Graphic Processor API

LIU Hui, TIAN Ze, LI Xiao-yu, CHEN Jia

(Aeronautical Computing Technique Research Institute, Xi'an 710119, China)

**Abstract:**3D graphic API (Application Programming Interface) is an abbreviation of 3D graphic processing function API. In order to make the application transplantable smoothly, it separates top application from bottom drivers in graphic system. OpenGL (Open Graphics Library) is defined to avoid different hardware drivers. How to develop the API which accords with OpenGL standard is the key point for research and application of 3D graphic processor. First analyze the attribute of OpenGL API, put forward the key technology for verifying self-researched API and OpenGL standard API. Verification is conducted from functional interface, drawing function, display function and error feedback on the basis of FPGA platform. The above method is easier to operate, more extensive coverage, more complex verifying examples. This method can accelerate the verifying process and improve the sufficiency of verification.

**Key words:**OpenGL kernel lib; accordance verification of standard; reliability of reading and writing; integrity of function; robustness

### 0 引 言

OpenGL 的全称是“开放的图形接口库函数”, 它是由 SGI 公司开发的一套标准的计算机图形处理接口, 它的实现方法与底层硬件无关, 屏蔽了不同硬件接口所带来的差异<sup>[1]</sup>, 对不同的操作系统提供了标准的编程接口, 使得应用程序在多操作系统下具有良好的移植性, 广泛应用于二维或三维图像的绘制。

3D 图形 API 库函数与 OpenGL 标准符合性验证是图形处理器开发的基础, 是图形处理器研制及应用

的关键, 是现代机载显示软件开发的前提与保障。

针对自研的 3D 图形处理器开发出符合 OpenGL 标准的 3D 图形 API, 是 3D 图形处理器研制及应用的关键。文中结合图形处理器研制与关键技术的研究, 首先分析了 OpenGL 标准的特点并将其归纳分类, 在此基础上提出了自研 3D 图形 API 验证的关键技术和方法, 涉及函数的读写属性、功能属性及错误信息反馈机制<sup>[2]</sup>, 最后在 FPGA 平台上进行了充分的验证, 结果表明自研 3D 图形 API 符合 OpenGL 标准。

收稿日期:2013-11-25

修回日期:2014-03-04

网络出版时间:2014-07-28

基金项目:“十二五”微电子预研(51308010601); 中国航空工业集团公司创新基金(2010BD63111)

作者简介:刘 晖(1986-), 男, 陕西宝鸡人, 硕士研究生, 研究方向为 SoC 设计与验证; 田 泽, 博士, 研究员, 中国航空工业集团首席技术专家, 研究方向为 SoC 设计、嵌入式系统设计、VLSI 设计等。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20140728.1222.016.html>

1 标准符合性验证

1.1 标准符合性验证流程

接口标准符合性验证是指以发布的标准为准则,对所设计的软件接口进行严格、定量、全面的验证,以确认产品是否符合相应的标准,或在多大程度上符合标准<sup>[3]</sup>。从验证方法学的角度来讲,标准符合性验证是标准的“黑盒”测试,即测试者完全不考虑接口的内部结构、实现和属性,只需考虑设计的验证用例是否能完全覆盖标准的每一款项<sup>[4-5]</sup>。标准的符合性验证过程如图 1 所示。

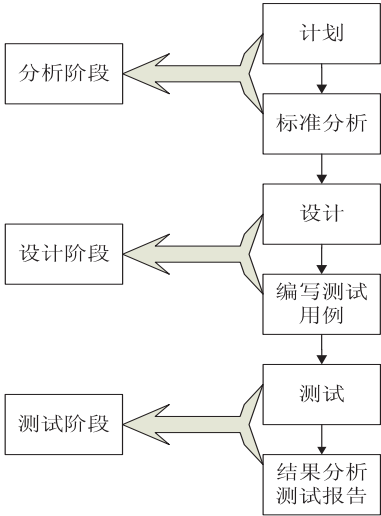


图 1 验证过程

1.2 标准符合性验证框架

OpenGL 标准 API 接口在图形处理器中位于 3D 引擎流水线处理当中,用于处理上层应用程序给底层图形处理器的绘图命令,最后需要将结果显示到屏幕上。按照标准规范的描述,文中将针对 OpenGL 标准 API 的符合性验证归纳为读写可靠性、功能完整性和鲁棒性<sup>[6-7]</sup>。

1.2.1 读写可靠性测试

可靠性衡量的是当接收到合法输入时,其输出与标准的符合程度,包括不同参数类型的输入和向量版本与非向量版本的函数<sup>[8]</sup>,对其验证项要做到全匹配。

OpenGL 中定义的数据类型一共有八种,分别与 C 语言中的数据类型对应,文中的验证要覆盖所有的符合函数定义的数据类型和函数类型。

例如:对 glColor() 函数有如下四种类型,每一类包括了字节型、短整型、整型、浮点型、双精度浮点型、无符号字节型、无符号短整型和无符号整型,函数如下所示:

```
void glColor3{b s i f ub us ui}( TYPE r, TYPE g, TYPE b );
void glColor3{b s i f ub us ui}v( const TYPE * v );
void glColor4{b s i f ub us ui}( TYPE r, TYPE g,
```

```
TYPE b, TYPE a );
void glColor4{b s i f ub us ui}v( const TYPE * v );
因此 glColor 函数验证项共有 4 * 7 = 28 项;对其每一项要设置相应的类型参数,读出写入的数据,查看数据的一致性,从而保证输出与标准参考有可比性。
```

1.2.2 功能完整性测试

功能测试又称为行为测试,根据标准的描述测试函数的特性和操作行为是否满足标准的描述<sup>[9]</sup>。功能测试只需要考虑各个功能点是否实现,而不用考虑整个软件的内部结构及代码,按照需求编写的测试用例,在实际结果和预期结果之间进行评测,进而更好地满足标准的需求。

例如:glBlendFunc( GLenum srcfactor, GLenum destfactor )。

如表 1 所示,分别给出了源片段和目标片段的混合参数,颜色的表示采用 RGB 表示法,alpha 表示饱和度信息。

表 1 glColorBlendFunc 参数设置表

srcfactor/ destfactor	RGB 混合因子	alpha 混合因子
GL_ZERO	(0,0,0)	0
GL_ONE	(1,1,1)	1
GL_SRC_COLOR	(Rs,Gs,Bs)	As
GL_ONE_MINUS_SRC_COLOR	(1,1,1)-(Rs,Gs,Bs)	1-As
GL_DST_COLOR	(Rd,Gd,Bd)	Ad
GL_ONE_MINUS_DST_COLOR	(1,1,1)-(Rd,Gd,Bd)	1-Ad
GL_SRC_ALPHA	(As,As,As)	As
GL_ONE_MINUS_SRC_ALPHA	(1,1,1)-(As,As,As)	1-As
GL_DST_ALPHA	(Ad,Ad,Ad)	Ad
GL_ONE_MINUS_DST_ALPHA	(1,1,1)-(Ad,Ad,Ad)	1-Ad
GL_CONSTANT_COLOR	(Rc,Gc,Bc)	Ac
GL_ONE_MINUS_CONSTANT_COLOR	(1,1,1)-(Rc,Gc,Bc)	1-Ac
GL_CONSTANT_ALPHA	(Ac,Ac,Ac)	Ac
GL_ONE_MINUS_CONSTANT_ALPHA	(1,1,1)-(Ac,Ac,Ac)	1-Ac
GL_SRC_ALPHA_SATURATE	(f,f,f);f=min(As,1-Ad)	1

验证过程如下:分别设置源片段和目标片段的混合参数,绘制出图形,并读出混合区域的颜色;利用源片段和目标片段的颜色与混合参数进行手动计算,计算出混合后的颜色值,将其与读出的混合颜色值相比较,查看功能的正确性。

因为源片段参数与目标片段参数分别有 15 种,因此共需验证 15 \* 15 = 225 项,才能保证功能验证的完整性。

1.2.3 鲁棒性测试

鲁棒性是指当函数接收到非法输入时,输出结果与标准的符合程度。当进行非法输入时,产生的错误信息由反馈环传输到应用层,并反馈给用户。规定的非法输入只产生错误信息,并不会中断程序的执行<sup>[10]</sup>。

鲁棒性测试包括:

- (1) 函数类型与参数类型不匹配;
- (2) 函数参数是一个不接受的枚举值;
- (3) 函数参数是定义域之外的值;
- (4) 代码语句不符合上下文规范。

例如: void glMap1 { fd } ( GLenum target, TYPE u<sub>1</sub>, TYPE u<sub>2</sub>, GLint stride, GLint order, const TYPE \* points );

标准中的错误项如下所示:

- (1) Target 不是一个可接受的值, 产生 GL\_INVALID\_ENUM;
- (2) u<sub>1</sub> 等于 u<sub>2</sub>, 产生 GL\_INVALID\_VALUE;
- (3) stride 小于一个控制点中值的数量, 产生 GL\_INVALID\_VALUE;
- (4) order 小于 1 或大于 GL\_MAX\_EVAL\_ORDER, 产生 GL\_INVALID\_VALUE;
- (5) 在 glBegin 和 glEnd 之间执行, 产生 GL\_INVALID\_OPERATION;
- (6) glMap1 被调用, 但是 GL\_ACTIVE\_TEXTURE 的值不是 GL\_TEXTURE0, 产生 GL\_INVALID\_OPERATION。

按照标准中规定的错误测试项来完成验证, 验证能否返回正确的信息; 对于标准中未定义的错误, 将在程序编译的过程中报错, 程序将不会继续执行。

## 2 验证模型

### 2.1 验证环境

环境模型是验证程序运行时所需要的硬件环境和软件环境。环境是验证的前提条件, 在验证之前要保证环境的安全性与连通性:

- (1) 在验证前搭建硬件环境, 保证测试环境能正常运行;
- (2) 设计验证用例时应考虑到对异常情况的处理要求, 对所有可能发生的异常情况都应该给出相应的处理措施;
- (3) 所有参数在使用前必须经过初始化, 防止参数越界, 发生不可预知的错误。

文中提出的验证方法的软件验证环境和硬件验证环境分别如表 2 和表 3 所示。

表 2 软件验证环境

软件名称	标识或版本	用途
VxWorks	5.5 及以上版本	操作系统
Tornado	2.2 及以上版本	测试程序编译和运行环境

### 2.2 程序模型

程序模型是指设计测试程序的整体结构和设计目

标。测试程序的整体结构包括: 初始值测试、读写测试、功能测试、显示测试、错误测试。其中的读写测试包括等价类范围测试和边界值测试, 功能测试和显示测试的结果与相同代码在 VS 平台下的绘图结果比较, 其他的测试项以 OpenGL 1.3 标准为准则。

表 3 硬件验证环境

硬件名称	标识和版本	数量	用途
PC 机	CPU: Inter Core2 Duo	1	编写测试计划, 设计测试用例, 阅读设计文档, 运行 Tornado 执行功能测试
	E7400 2.80 GHz		
	内存: 1.99 GB		
	系统: Windows XP SP3		
显示器	电脑显示器	1	显示输出的图形
FPGA 芯片	XC6VLX760	1	搭建测试环境
PPC	PPC8270	1	搭建测试环境

测试程序模型模块划归如下所示:

```
void Draw( void)
{
/* * * * * * 初始值测试模块 * * * * * /
默认状态值测试;
默认数据值测试;
/* * * * * * 数据读写测试模块 * * * * * /
状态读写测试;
数据读写测试;
/* * * * * * 功能测试模块 * * * * * /
功能测试;
/* * * * * * 显示测试模块 * * * * * /
显示测试;
/* * * * * * 错误测试模块 * * * * * /
错误测试;
}
```

其中, 接口参数测试包括了初始值测试模块和数据读写测试模块, 对应于标准符合性测试关键点的读写可靠性测试; 绘图功能测试和显示功能测试对应于标准符合性测试关键点的功能完整性测试; 错误反馈处理对应于标准符合性测试关键点的鲁棒性测试<sup>[11]</sup>。

## 3 OpenGL 标准符合性验证过程

### 3.1 测试函数排序

按照功能将 OpenGL1.3 函数库中的函数分模块划分, 把基本的功能缓冲区验证、基本操作验证和基本图元验证放在最前面, 这类操作关联的资源相对较少, 而且是后面功能验证的基础, 必须要先保证其正确性; 第二部分是堆栈操作、变换操作、显示操作和像素相关的操作; 最后一部分是一些功能的验证, 包括光照和材质、纹理操作、求值器、显示列表; 按照这个顺序, 验证项将实现由初级向高级, 由简单到复杂, 由单一化到功能化的过渡, 保证了资源验证的正确性和合理性。

3.2 验证方案

在详细地分析了标准之后,归纳出了需要验证的模块和验证用例的设置:

- (1)参照 OpenGL1.3 函数库对函数单元的初始值、功能、接口、边界、错误处理等应逐项进行验证;
- (2)每个函数特性应至少被一个正常验证用例和一个被认可的异常验证用例覆盖;
- (3)验证用例的输入应至少包括有效等价类值、无效等价类值和边界数据值<sup>[12]</sup>;等价类是指某个输入域的子集合。在该子集合中,各个输入数据对于揭示程序中的某一个功能都是等效的。因此,可以用等价类中的某一个值的验证结果代表等价类中所有值的验证结果。

3.3 验证步骤

以验证方案为指导,设计每一个函数的验证过程如下所示:

- (1)编写各个单元的验证用例,包括验证用例函数标识、函数功能的描述、参数设置、相关错误信息、测试目的、测试方法、输入和预期输出等;
- (2)根据编写的验证用例,在测试环境中设置相应的初始化条件,验证参数的默认值和默认值的输出结果是否与预期的结果相等;
- (3)设置参数进行接口验证;
- (4)若参数变化在一个范围内,需要验证有效参数类,无效参数类和边界数据值;若参数为一个枚举形变量,需要验证到所有的参数<sup>[13]</sup>;
- (5)对于每一类的输入,应该明确给出返回值或输出的信息;
- (6)设置参数进行功能验证;
- (7)每个验证函数,必须包含一个正常的验证用例和一个被认可的异常验证用例;
- (8)如果实测结果与预期结果不符,在确认证验环境和用例没有问题后,在过程管理平台填写函数验证的问题,并与项目开发人员进行沟通;
- (9)提出问题后,继续跟踪遗留问题的解决,并根据设计和代码的修改情况完成回归验证。

3.4 验证评价

依据测试程序的整体结构,可将验证项的评价分为四类:

- (1)默认值验证:将读出的默认值与标准对照;
- (2)读写验证:将参数赋值为符合类型的数据,读出参数数据并与所赋值相比较;
- (3)功能与显示验证:对于改变矩阵数据的函数,分别读出改变前后的矩阵数据,再将手动计算的矩阵值与其相比较<sup>[14]</sup>;对于可显示的函数,首先要确定数据的正确性,其次要判断实现功能的优略,将显示结果

与相同代码在 VS 平台下的显示结果进行比较而确定;

- (4)非法值验证:对于已定义的参数被非法使用,编译过程不会报错,这将通过设计的反馈环机制向用户发送错误信息提示;对于未被定义的参数或不符合函数定义的参数,编译过程将提示错误。

3.5 验证结果

该验证过程在 FPGA(XC6VLX760)原型平台下进行,将函数类型与参数类型做笛卡尔积,对所得的验证用例逐个验证。结果表明,所设计的 OpenGL 函数接口的默认值验证、读写验证、功能验证和非法项验证都与标准相符合。FPGA 平台下与 VS 平台下所绘制图形的比较效果如图 2 所示,可以看出所绘制的图形用肉眼几乎看不出区别。但是该验证结果只能证明单条指令执行的正确性,不能证明组合指令功能模块的正确性,其还与指令流水线和数据流水线的设计相关。

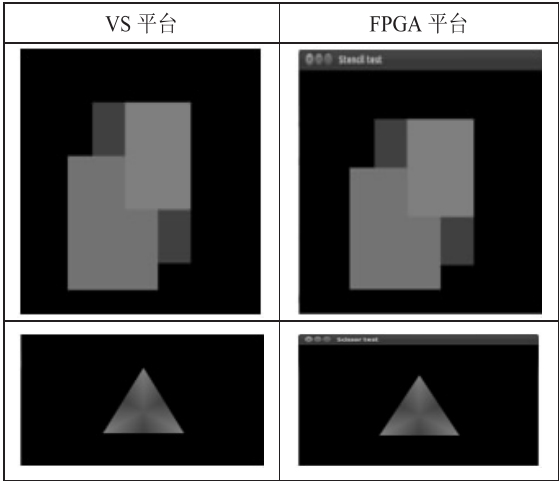


图 2 显示效果对比

4 结束语

随着图形处理器的发展,其内部的逻辑部件功能越来越强大,实现越来越复杂,但是要实现的核心接口函数基本不变。文中以验证 OpenGL1.3 版本的 API 函数为目标,设计了基于标准符合性的验证流程与验证框架,用以规范化验证过程,设计的验证方法覆盖了函数的各个功能与参数,实现了函数验证的完整性。

我国的标准化规范制定与标准验证的工作起步较晚,目前尚无主导的符合性验证标准被其他国家使用,标准化工作的质量有待进一步的提高。文中所提出的验证方法对国内同类型的标准验证有一定的推广价值及借鉴意义。

参考文献:

[1] Shreiner D, The Khronos OpenGL ARB Working Group. Open-  
(下转第 201 页)



据与回波强度之间的数学模型,利用深度数据对声纳图像作修正,提高图像的准确性。除此之外,改善声纳图像的灰度不均匀也是有待深入研究的问题。

参考文献:

[1] 唐秋华,陈义兰,周兴华,等.多波束海底声像图的形成及应用研究[J].海洋测绘,2004,24(5):9-12.

[2] Brown C J, Blonde P. Developments in the application of multibeam sonar backscatter for seafloor habitat mapping [J]. Applied Acoustics,2009,70(10):1242-1247.

[3] 丁继胜,刘忠臣,周兴华,等.多波束测深声纳系统的工作原理[J].海洋测绘,1999(3):15-22.

[4] Keeton J A,Searle R C. Analysis of simrad EM12 multibeam bathymetry and acoustic backscatter data for seafloor mapping,exemplified at the mid-Atlantic ridge at 45°N[J]. Marine Geophysical Researches,1996,18(6):663-688.

[5] Chenadec G L,Boucher J M. Sonar image segmentation using the angular dependence of backscattering distributions [C]//Proc of oceans. [s.l.]:IEEE,2005:147-152.

[6] 赵建虎,刘经南.多波束测深及图像数据处理[M].武汉:武汉大学出版社,2008.

[7] 唐秋华,周兴华,丁继胜,等.多波束反向散射强度数据处理研究[J].海洋学报,2006,28(2):51-55.

[8] Mitchell N C. Processing and analysis of simrad multibeam sonar data[J]. Marine Geophysical Researches,1996,18:729-739.

[9] Urick R J. Principles of underwater sound for engineers[R]. [s.l.]:Mc Graw-Hill,1967.

[10] 张会娟,阳凡林,李峰,等.多波束声纳图像整体灰度不

均匀及中央波束区反射异常校正方法[J].全球定位系统,2012,37(1):25-27.

[11] 王煜.多波束声纳图像入射角效应和镜面反射区异常的改正[D].青岛:山东科技大学,2011.

[12] 马飞虎,鄂栋臣,赵建虎,等.多波束声强振幅滤波方法比较研究[J].武汉大学学报:信息科学版,2010,35(9):1082-1085.

[13] 孙文川,肖付民,金绍华,等.加权最小二乘估计的多波束声强数据归一化方法[J].测绘科学技术学报,2012,29(4):262-266.

[14] 田晓东.声纳图像滤波方法的比较分析[J].声学与电子工程,2007(1):22-25.

[15] 朱小辰,刘雁春,夏伟,等.基于回波时间的多波束测深与声纳数据匹配方法[J].海洋测绘,2010,30(1):25-28.

[16] Lurton X, Dugelay S, Augustin J M. Analysis of multibeam echo-sounder signals from the deep seafloor [C]//Proc of oceans engineering for today's technology and tomorrow's preservation. Brest:IEEE,1994:213-218.

[17] Buelens B, Williams R, Sale A, et al. Model inversion for mid-water multibeam backscatter data analysis [C]//Proc of oceans. [s.l.]:IEEE,2005:431-435.

[18] Becker K. Effect of various surface-height-distribution properties on acoustic back-scattering statistics[J]. IEEE Journal of Oceanic Engineering,2004,29(2):246-259.

[19] 金绍华,翟京生,刘雁春,等.Simrad EM多波束反向散射强度数据精处理研究[J].测绘科学,2010,35(2):106-108.

[20] 郝燕玲,韩庆楠,徐聪.伪灰度双模板匹配的多波束数据拼接算法[J].哈尔滨工程大学学报,2011,32(7):927-931.

(上接第 196 页)

GL programming guide[M].李军,徐波,译.第7版.北京:机械工业出版社,2010.

[2] Chen Bo, Cheng H H. Interpretive OpenGL for computer graphics[J]. Computers & Graphics,2005,29(3):331-339.

[3] Baek N, Lee Hwanyong. OpenGL ES 1.1 implementation based on OpenGL[J]. Multimedia Tools and Applications,2012,57(3):669-685.

[4] Al-Salman A S, Al-Qahtani F. GLAL:an OpenGL arabization library[J]. Software Practice and Experience,2005,35(14):1393-1407.

[5] 许文馨,左昕,安琪,等.标准符合性测试的国内外比较及对策分析[J].现代商贸工业,2013,25(1):155-157.

[6] 向润.黑盒测试方法探讨[J].软件导刊,2009,8(1):33-34.

[7] 赵瑞莲.软件测试方法研究[D].北京:中国科学院研究生

院(计算技术研究所),2001.

[8] 杨建光,梅大成.黑盒测试方法和综合策略的研究[J].计算机光盘软件与应用,2012(4):121-122.

[9] 石君友,纪超,李海伟.测试性验证技术与应用现状分析[J].测控技术,2012,31(5):29-32.

[10] 刘新建.标准验证工作的思考[J].中国标准化,2012(1):81-84.

[11] 马城城,田泽,黎小玉.基于GPU FPGA芯片原型的Vx-Works下驱动软件开发[J].计算机技术与发展,2013,23(7):84-86.

[12] 李炎琨.基于VMM验证方法的研究[D].西安:西安电子科技大学,2010.

[13] 伍小荟.面向软件可靠性的性能测试方法研究与应用[D].上海:上海交通大学,2012.

[14] 夏启明.软件测试及评价的复用策略研究及其实现[D].武汉:武汉大学,2010.

### 3 D图形处理器API符合性验证方法关键技术研究

作者：[刘晖](#)，[田泽](#)，[黎小玉](#)，[陈佳](#)，[LIU Hui](#)，[TIAN Ze](#)，[LI Xiao-yu](#)，[CHEN Jia](#)  
作者单位：[中航工业西安航空计算技术研究所, 陕西 西安, 710119](#)  
刊名：[计算机技术与发展](#)   
英文刊名：[Computer Technology and Development](#)  
年，卷(期)：2014(10)

本文链接：[http://d.wanfangdata.com.cn/Periodical\\_wjfz201410048.aspx](http://d.wanfangdata.com.cn/Periodical_wjfz201410048.aspx)