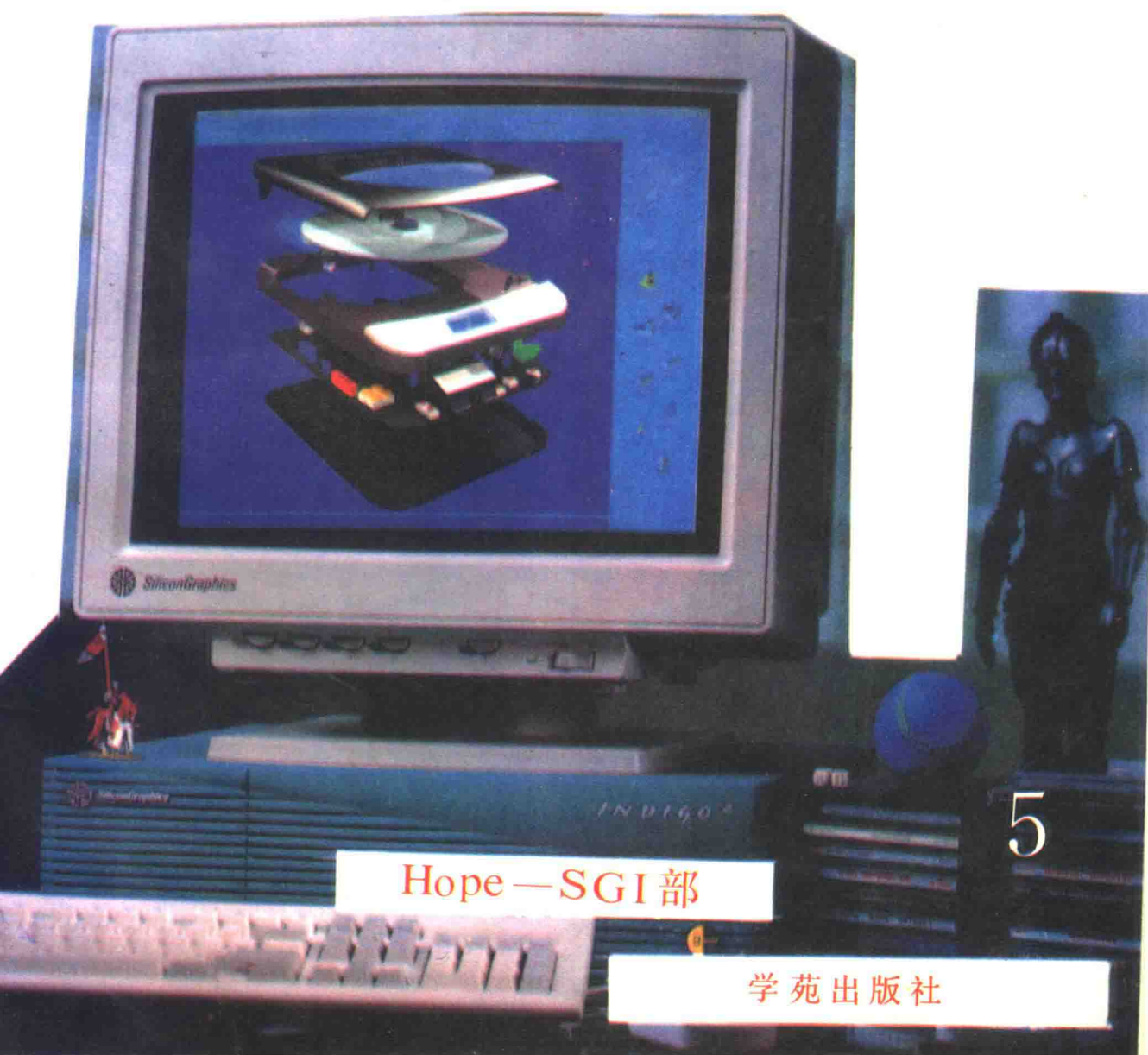


GL 参考手册

FOTRAN 版

曹 康 李 娟 编



Hope — SGI 部

学苑出版社

5

微机图形图像与 CAD 系列丛书

GL 参考手册

FORTRAN 版

曹 康 李 娟 编写

学苑出版社

1993.

(京)新登字 151 号

内 容 提 要

美国 SGI(Silicon Graphic)公司生产的 IRIS 图形工作站以其卓越的图形性能及标准化的开发环境倍受广大用户的青睐,在九十年代工作市场独领风骚。

SGI 图形工作站采用图形流水线机制,彻底解决了 shading、消隐以及 NURBS 曲面显示等一系列图形学难题,实现了真三维显示,是新一代图形工作站的标志。

我国从八十年代末开始引进 SGI 工作站,用户数量迅速增加。SGI 工作站已成为各行业尤其是 CAD、可视计算和动画制作的优先机种。为了广大用户的使用方便,我们将部分 SGI 工作站资料编成中文资料,以供读者选用。

欲购本书的用户,请直接与北京 8721 信箱联系,电话 2562329,邮码 100080。

微机图形图像与 CAD 系列丛书

GL 参考手册 FORTRAN 版

编 写:曹 康 李 娟

责任编辑:甄国宪

出版发行:学苑出版社 邮政编码:100032

社 址:北京市西城区成方街 33 号

印 刷:兰空印刷厂

开 本:787×1092 1/16

印 张:26.125 字 数:594 千字

印 数:1~5000 册

版 次:1993 年 12 月北京第 1 版第 1 次

ISBN7-5077-0802-0/TP·13

本套定价 300.00 元(五册/套)

学苑版图书印、装错误可随时退换

FORTTRAN 语言图形库参考指南(4.0)版

intro

函数名

intro——该函数用于描述图形库和分布式图形库中的各函数。

++++++
+ 说 明 +
++++++

本手册是图形库(GL)和分布式图形库(DGL)函数参考指南。若需 GL 和 DGL 更详细的介绍,可参看《GL 编程指南》。

一般来说,DGL 支持 GL 中的所有函数。但对某些函数仍存在细微差别。此外,一些函数(dglopen)和(dglclose)在 DGL 中可以运行,但在 GL 中则不行。对于函数的这种差别,本手册将给予说明。

本手册中每页的内容可以联机获取。若需查阅,可以使用 IRIX 命令:

man routine name <Enter>

由于这些手册页是用于联机参考,因此要求每页内容要简捷。每一页均由以下几部分组成:

函数名

列出所说明的函数的名称。

++++++
+ FORTRAN. 格式 +
++++++

列出函数的类型说明及参数。

参 数

说明该函数所用的参数。

函数返回值

说明一个函数的返回值。

描 述

介绍如何使用该函数。

参 见

列出与该函数有关的函数及其它信息源。

举 例

举例说明如何使用该函数。

注 释

着重指出有关该函数的各种限制情况。

缺 点

说明与设定功能的偏差以便在新版本中改进。

头文件

在/usr/include/gl中有三个头文件应包含在调用图形库函数的源文件中。这些文件是fgl.h, fget.h, fdevice.h。

图形库函数清单

acbuf——操作累加缓冲器
acsize——定义累加缓冲器中每个颜色成份的位面数
addtop——向现存弹出菜单中添加菜单项目
afunct——定义 alpha 测试函数
arc,arci,arcs——绘圆弧
arcf,arcfi,arcfs——绘填充圆弧
attach——将光标连到两个取值器
backbu——启用和停用后台缓冲器绘图
backfa——打开或关闭背向多边形消隐方式
bbox2, bbox2i, bbox2s——选取或删除边界框及最小像素半径。
bgnclo——开始定义一条闭合线段的顶点
bgnlin——开始定义一条线段的顶点
bgnpoi——开始定义点
bgnpol——开始定义多边形顶点
bgnqst——开始定义四边形带顶点
bgnsur——开始定义一个 NURBS 曲面
bgntme——开始定义三角形网格顶点
bgntri——开始定义 NURBS 曲面裁剪环线
blanks——控制屏幕开关
blankt——设置导致屏幕关闭的等待时间
blendf——计算一个像素的混合颜色值
blink——以可选速率改变颜色表项
blkqre——从队列中读取多个项
c3f,c3i,c3s,c4f,c4i,c4s——指定当前颜色向量的 RGB(或 RGBA)值
callob——绘物体实例
charst——在屏幕上绘一串光栅字符
chunks——定义内存中的最小物体尺寸
circ,circi,circs——绘出一个圆形轮廓
circf,circfi,circfs——绘一个填充圆
clear——清除视区
clearh——将全程变量置零
clippl——设定一个平面用于裁剪各种几何形体
clkon,clkoff——控制键盘按动

closeo——结束一个物体定义
 cmode——将颜色表方式置为当前方式
 cmov,cmovi,cmovs,cmov2,cmov2i,cmov2s——更新当前字符位置
 color,colorf——设置当前绘图方式下的颜色号
 compac——压缩物体存储
 concav——允许系统绘凹多边形
 cpack——用一个单 32 位整数定义 RGBA 颜色
 crv——绘曲线
 crvn——绘一系列曲线段
 curori——定义光标原点
 curson,cursof——控制光标在当前窗口的可见性
 cursty——定义光标类型和/或尺寸
 curveb——选择绘曲线的基矩阵
 curvei——绘曲线段
 curvep——设置用于绘曲线段的线段数
 cyclem——以给定速率循环使用颜色表
 czelea——同时清颜色位面和 Z 缓冲器
 dbtext——设置拨号和按钮盒正文显示
 defbas——定义一个基矩阵
 defcur——定义光标图案
 deflin——定义线型
 defpat——定义模式
 defras——定义光栅字型
 delobj——删除一个物体
 deltag——从当前打开物体中删除标识符
 depthc——打开和关闭深度暗示方式
 dglclo——关闭 DGL 服务器联接
 dglope——打开 DGL 服务器联接
 dopup——显示指定的弹出菜单
 double——置显示方式为双缓冲器方式
 draw,drawi,draws,draw2,draw2i,draw2s——绘线
 drawmo——选择绘图用 GL 帧缓冲器
 editob——打开物体定义用于编辑
 endclo——闭合线顶点定义结束
 endfee——控制反馈方式
 endful——结束全屏幕方式
 endlin——线段顶点定义结束
 endpic——关闭拾取方式

endpoi——绘点集的顶点定义结束
 endpol——多边形顶点定义结束
 endqst——四边形带顶点定义结束
 endsel——关闭选择方式
 endsur——NURBS 曲面定义结束
 endtme——三角形网格顶点定义结束
 endtri——定义一条 NURBS 曲面裁剪环线定义结束
 feedba——控制反馈方式
 finish——封锁几何流水线直至为全空
 fogver——为每个顶点的大气效应设定雾密度
 font——为绘正文字符串选择光栅字型
 foregr——防止将一个绘图进程转入后台
 freepu——释放一个菜单
 frontb——启用和停用前台缓冲器绘图
 frontf——关闭和打开前向多边形消隐
 fudge——定义图形窗口的初始尺寸
 fullsc——设置全屏幕写状态
 gammar——定义用于 γ 修正的颜色表变化率
 gbegin——建立全屏幕窗口
 gconfi——系统重新配置
 genobj——返回一个整数用作物体标识符
 gentag——返回一个用整数用作标识符
 getbac——返回是否显示背向多边形
 getbuf——确定使用的缓冲器
 getbut——返回按钮状态
 getcmm——返回当前颜色表方式
 getcol——返回当前颜色
 getcpo——返回当前字符位置
 getcur——返回光标特性
 getdem——指定打开或关闭深度暗示方式
 getdep——舍弃函数
 getdes——返回字符特性
 getdev——一次读出一系列取值器
 getdis——返回当前显示方式
 getdra——返回当前绘图方式
 getfon——返回当前光栅字型号
 getgde——获取图形系统说明
 getgpo——获取当前图形位置

gethei——返回当前光栅字形中的最大字符高度
gethit——返回当前全程变量值
getlsb——当前系统无此函数
getlsr——返环线型重复数
getlst——返回当前线型
getlwi——返回当前线宽
getmap——返回当前颜色表号
getmat——返回变换矩阵的备份
getmco——获取用于颜色表项的 RGB 值备份
getmmo——返回当前矩阵方式
getmon——返回当前显示器类型
getnur——返回被裁剪的 NURBS 曲面显示特性的当前值
getope——返回当前打开物体的标识符
getori——返回图形窗口位置
getpat——返回当前模式号
getpla——返回可用位面数
getres——返环线型设置方式状态
getsb——读取当前计算屏幕界盒
getscr——返回当前屏幕屏蔽
getsiz——返回图形窗口尺寸
getsm——返回当前着色模型
getval——返回某个取值器的当前状态
getvid——获取视频硬件寄存器
getvie——获取当前视区尺寸
getwri——返回当前写屏蔽码
getwsc——返回当前窗口出现的屏幕
getzbu——返回 Z 缓冲开关状态
gexit——退出图形库系统
gflush——清理 DGL 客户缓冲器
ginit——建立全屏幕窗口
glcomp——控制兼容方式
greset——重新设置图形状态
gRGBco——获得当前 RGB 颜色值
gRGBma——返回当前 RGB 写屏蔽码
gselec——将系统置于选择方式
gsync——等待垂直回扫周期
gversi——返回图形硬件和图形库版本信息
iconsiz——指定窗口图符尺寸

conti——指定当前图形窗口的图符名称
 imakeb——登记屏幕后台进程
 initna——初始化名称栈
 isobj——测试某物体是否存在
 isqueu——测试指定设备是否进入队列
 istag——返回当前物体标识符
 keepas——定义图形窗口纵横比
 lampon,lampof——控制键盘指示灯
 linesm——指定直线反走样
 linewi——指定线宽
 lmbind——选择新材料、光源和光照模型
 lmcolo——光照激活时改变颜色命令效应
 lmdef——定义或修改材料、光源或照明模型
 loadma——调入变换矩阵
 loadna——将一个名称调入名称栈
 logico——指定象素写的逻辑运算
 lookat——定义观察变换
 lrectr——将一矩形象素阵列读入 CPU 内存
 lrectw——将一矩形象素阵列写到帧缓冲器
 lRGBra——定义用于深度暗示的 RGB 颜色范围
 lsback——控制线段末端是否加色
 lsetde——定义深度范围
 lshade——定义用于深度暗示的颜色号范围
 lsrepe——当前线型重复因子
 makeob——创建一个物体
 maketa——对显示序列中的函数进行计数
 mapcol——修改颜色表项
 mapw——将屏幕上一个点映射成三维世界坐标中的一条直线
 mapw2——将屏幕上一个点映射到二维世界坐标
 maxsiz——定义图形窗口最大尺寸
 minsiz——定义图形窗口最小尺寸
 mmode——定义当前矩阵方式
 move,movei,moves,move2,move2i,move2s——将当前图形位置移到指定点
 mswapb——同时交换多个帧缓冲器
 multim——将颜色表分解为多个更小的表
 multma——左乘当前转换矩阵
 n3f——指定一个法矢
 newpup——为新菜单分配并初始化一个结构

newtag——在与现存标识符相关的物体内建立新的标识符
 nmode——重新定义法矢
 nobord——定义一个无边框窗口
 noise——取值器运动过滤
 noport——指定一个程序不占屏幕空间
 nurbsc——控制 NURBS 裁剪曲线形状
 nurbss——控制 NURBS 曲面的形状
 objdel——从一个物体删除函数
 objins——将函数插入到一个物体的指定位置
 objrep——更新现存显示序列函数
 onemap——将颜色表组织成一幅大颜色表
 ortho,ortho2——定义正交投影变换
 overlay——分配上覆位面
 Pageco——设置正文区背景颜色
 passth——通过几何管道传递单一标记
 patch——绘一个曲面片
 patchb——定义当前基矩阵
 patchc——定义用于表达一块曲面片用的曲线数
 patchp——定义在一个曲面片上绘曲线的精度。
 pclos——结束一个填充多边形
 pdr,pdri,pdrs,pdr2,pdr2i,pdr2s——定义多边形的下一个顶点
 perspe——定义透视投影变换
 pick——将系统置为拾取方式
 picksi——定义拾取区域范围
 pixmod——定义像素变换方式参数
 pmv,pmvi,pmvs,pmv2,pmv2i,pmv2s——定义多边形的第一点
 pnt,pnti,pnts,pnt2,pnt2i,pnt2s——绘点
 pntsmo——指定点反走样
 polarv——定义观察者在极坐标中的置
 polf,polfi,polfs,polf2,polf2i,polf2s——绘填充多边形
 poly,polyi,polys,poly2,poly2i,poly2s——绘多边形轮廓
 polymo——控制多边形显示方式。
 polysm——定义多边形反走样
 popatt——弹出属性栈
 popmat——弹出变换矩阵栈
 popnam——将一个名称弹出名称线
 popvie——弹出视区栈
 prefpo——定义图形窗口的优选位置和尺寸

prefsi——定义图形窗口优选尺寸
 pupmod——舍弃函数
 pushat——下压属性栈
 pushma——下压变换矩阵栈
 pushna——将一个新名称压入名称栈
 pushvi——下压视区栈
 pwlcu——定义 NURBS 曲面的分段线性裁剪曲线
 qdevic——使设备进入队列
 qenter——建立事件队列项
 qgetfd——返回事件队列的文件描述符
 qread——读事件队列中的第一项
 qreset——置事件队列为全空
 qtest——检查事件队列内容
 rcrv——绘一条有理曲线
 rcrvn——绘一系列曲线段
 rdr,rdri,rdrs,rdr2,rdr2i,rdr2s——相对绘图
 readpi——返回指定像素的像素值
 readRG——获取指定像素的像素值
 readso——设置各种函数均可读的像素源
 rect,recti,rects——绘出矩形区域轮廓
 rectco——按可选的缩放比例拷贝一个矩形的像素
 rectf,rectfi,rectfs——填充矩形区域
 rectre——将一个矩形的像素读入 CPU 内存
 rectwr——将一个矩形的像素写入帧缓冲器
 rectzo——设置矩形像素拷贝和写的缩放比例
 resetl——控制线型的连续性
 reshapi——将视区大小置为当前图形窗口
 RGBcol——以 RGB 方式定义当前颜色
 RGBmod——用 RGB 颜色方式进行显示和真实显示
 RGBran——舍弃函数
 RGBwri——置可见位置写屏蔽码
 ringbe——键盘振铃
 rmv,rmvi,rms,rmv2,vmv2i,rmv2s——相关移动
 rotate,rot——旋转图形元素
 rpatch——绘一有理曲面片
 rpdr,rpdr2,rpdr2i,rpdr2s——相关多边形绘图
 rpmv,rpmvi,rpmvs,rpmv2,rpmv2i,rpmv2s——相关多边形移动
 sbox,sboxi,sboxs——绘有屏幕对齐的矩形

sboxf,sboxfi,sboxts——绘填充的与屏幕对齐的矩形
 scale——对物体进行比例和镜象变换
 sclear——将模板位面复位到设定值
 scrbox——控制屏幕框
 screen——将世界坐标映射到屏幕绝对坐标系
 scrmas——定义矩形屏幕裁剪屏蔽
 scrnat——为屏幕连输入焦点
 scrnse——选取放置新窗口的屏幕
 scrsub——在屏幕空间范围内划分线和多边形
 setbel——定义键盘振铃持续时间
 setcur——定义光标特性
 setdbl——设置拨号盒和按钮盒灯光
 setdep——舍弃函数
 setlin——选取线型模式
 setmap——选取一种由多颜色表方式提供的小颜色表
 setmon——设置监视器类型
 setnur——定义裁剪的 NURBS 曲面的显示特性
 setpat——选择多边形和矩形的填充模式
 setpup——定义一个给定弹出菜单项的显示特性
 setsha——舍弃函数
 setval——指定取值器的初始值和范围
 setvid——设置视频硬件寄存器
 shadem——选择着色模型
 shader——舍弃函数
 single——存显所有位面
 smooth——舍弃函数
 spelos——舍弃函数
 splf,splfi,splfs,splf2,splf2i,splf2s——绘着色填充多边形
 stenci——改变模板操作参数
 stensi——定义作为模板面的位面数
 stepun——指定图形窗口以离散步长改变尺寸
 strwid——返回已定义的正字符串宽度
 subpix——控制点、线、多边形顶点的放置
 swapbu——交换标准帧缓冲器的前后台缓冲器
 swapin——定义缓冲器转换最小时间
 swaptm——开关三角形网格寄存器指针
 swinop——建立图形子窗口
 swrite——设置可写模板位
 t2d,t2f,t2i,t2s——指定纹理坐标
 tevbin——选择纹理环境
 tevdef——定义纹理映射环境
 texbin——选择纹理函数

texdef——将二维图象转变为纹理
 texgen——指定纹理坐标的自动生成
 textco——定义正文区的正文颜色
 textin——初始化正文区
 textpo——设置正文区位置和尺寸
 tie——把两个取值器连到一个按钮上
 tpon, tpoff——控制正文区可见性
 transl——平移图形元素
 underl——分配下伏颜色显示位面
 unqdev——终止指定设备在事件队列中列项
 v2d, v2f, v2i, v2s, v3d, v3f, v3i, v3s, v4d, v4f, v4i, v4s——将二维, 三维或四维顶点输
 送到图形管道中。
 videoc——初始化一个任选图象外设的指令传输序列
 winatt——舍弃函数
 viewpo——为一个图象分配一块窗口区
 winclo——关闭图形窗口
 wincon——为当前窗口设置窗口约束
 windep——测定窗口在栈中的位置
 window——定义透视投影变换
 winget——返回当前图形窗口标识符
 winmov——将当前图形窗口移到左下角
 winope——建立图形窗口
 winpop——将当前窗口移至其它所有窗口之上
 winpos——改变当前图形窗口位置和大小
 winpus——将当前窗口放至所有其它窗口之后
 winset——指定当前图形窗口
 wintit——为当前窗口附加标题栏
 wmpack——用单整数定义 RGBA 写屏蔽
 writem——设置位面写权限
 writep——在屏幕上为一排像素上色
 writeR——在屏幕上为一排像素上色
 xfpt, xfpti, xfpts, xfpt2, xfpt2i, xfpt2s, xfpt4, xfpt4i, xfpt4s——用反馈方式使一点
 和当前矩阵相乘
 zbuffe——启用或停用 Z 缓冲器在当前帧缓冲器中的运行
 zclear——初始化当前帧缓冲器的 Z 缓冲器
 zdraw——启用或停用向 Z 缓冲器存取图形
 zfunct——指定 Z 缓冲器比较函数
 zsourc——选取 Z 缓冲器比较源
 zwrite——为当前帧缓冲器的 Z 缓冲器设置写屏蔽码

acbuf

函数名

acbuf——操作累加缓冲器。

* FORTRAN 格式 *

```
subroutine acbuf(op, value)
integer*4 op
real value
```

参 数

op 是以下六个符号常量之一:

AC CLEAR: 将红、绿、蓝和 alpha 累加缓冲器的值全部置为 *value* (圆整到最近整数)。 *value* 的取值范位为 16 位符号整数。

AC ACCUMULATE: 从当前 readso 库中取像素。它们的红、绿、蓝和 alpha 分量按 *value* 进行比例转换。所得各分量的 16 位像素元加到累加缓冲器的已有像素中。 *value* 的范围是 -255.996 至 255.996。超出这个范围取 *value* 的最大绝对值。累加值不会被固定到累加缓冲器的 16 位符号整数范围内,因而只有通过限制累加操作范围来避免溢出。

AC CLEAR ACCUMULATE: 这是一有效的组合指令,其作用是首先将累加缓冲器清零,然后按每个 AC ACCUMULATE 进行累加,累加范围和截断值由每个 AC ACCUMULATE 确定。

AC RETURN: 从累加缓冲器取像素。其红、绿、蓝和字符分量按 *value* 值进行比例转换。所得到的 8 位像素分量被写到当前启用的绘图缓冲器(前台、后台、Z 缓冲器)中。在这种转变过程中所有特殊的像素操作(Z 缓冲器、混合函数、逻辑运算、模板、纹理映射等)将被忽略。目标值被简单取代。然而这种操作要受当前视区和屏幕掩码的限制。 *value* 值的范围为 0.0 到 1.0。超过这个范围的值将超出部分截去。按 *value* 转换后,

颜色分量在被写到当前绘图缓冲器前其取值前用为 0~255。

AC MULT: 每一个累加缓冲器像素的红、绿、蓝和字符分量按 *value* 作比例变换。

AC ADD: 将 *value* 值附加到累加缓冲器中每个像素的每个红、绿、蓝和字符分量上。

value: *value* 值是一个浮点数,如何使用 *value* 由 op 决定

描 述

累加缓冲器是一个 64 位的像素库,红、绿、蓝和 alpha 分量各占十六位,与屏幕像素一一对应。存储在标准帧缓冲器中的像素图象(一般由几何数据生成)可以被累加到累加缓冲器。这些像素在传输过程中是由一个范围和精度被限定的浮点数进行转换。此后,经过累加的图象可以被送回到标准帧缓冲器。

诸如点、线和多边形的光滑化效应、模糊移动效应、景深效应等都可以用累加图象来创建,这些累加映像可以用不同的转换矩阵生成。只有当子像素代码为“直 TRUE”时(参见 subpix)才能实现。readso 方式可以与其它像素读操作共享,包括 rectre 和 rectco。但 rectz00m 对累加操作没有影响。

所有累加缓冲器操作均受前视区视区的屏幕码限制。

累加缓冲器是标准帧缓冲器的一部分。只有当绘图方式处于 NORMALORAW(标准绘图方式)时以及标准帧缓冲器处于 RGB 方式时才能调用 acbuf。

参 见

acsize, drawmo, subpix, scrmas。

注 释

当 acsize 值为零时如果调用累加操作将报告出错信息,并终止任何操作。

acsize

函数名

acsize——定义累加缓冲器中每个颜色单元的位面数。


```
*****
*  FORTRAN  格式  *
*  *****
```

```
subroutine acsize(planes)
integer*4 planes
```

参 数

planes: planes 设定为累加缓冲器中每个颜色单元保留的位面数。可接受值为 0 (缺省值)和 16。

描 述

所显示的图像是按每个颜色单元不大于 8 位的方式被累加到(参见 acbuf)帧缓冲器中。为了重新设定容量,必须在 acsize 之后调用 gconfig。

累加缓冲器容量的缺省值是 0,这意味着停止进行图像累加。

每个分量 16 位的累加缓冲器是带符号的,因此它所支持的累加值范围为 -32768~32767。

参 见

acbuf, drawmo, gconfi。

注 释

该函数只能以立即方式调用。

累加缓冲器只有在标准帧缓冲器中才能调用。只有当绘图方式处于“标准状态(NORMAL-DRAW)”时才能调用 acsize。

addtop

函数名

addtop——向现存弹出菜单中添加菜单项目。