

寄存器	地址	读/写	描述	复位值
FGVS_CINT	0x72018000 ~0x7201803F	读/写	顶点着色器的常量整数寄存器	0xFFFFFFFF

字 0 (0x72018XX0)

	位	描述	复位值
W	[31:24]	常量整数 W 组成部分值	0xX
Z	[23:16]	常量整数 Z 组成部分值	0xX
Y	[15:8]	常量整数 Y 组成部分值	0xX
X	[7:0]	常量整数 X 组成部分值	0xX

7. 常量布尔寄存器

常量布尔值可以储存在常量布尔寄存器内。常量布尔值只用于静态流量控制。常量布尔寄存器是 16 位的布尔寄存器。寄存器序号与每个位位置相对应。TRUE 用 1 表示，FALSE 用 0 表示。

寄存器	地址	读/写	描述	复位值
FGVS_CBOOL	0x72018400	读/写	顶点着色器的常量布尔寄存器	0xFFFFFFFF

CBOOL	位	描述	初始状态
Reserved	[31:16]	保留	0xFFFF
REG15	[15]	常量布尔寄存器 15	X
REG14	[14]	常量布尔寄存器 14	X
REG13	[13]	常量布尔寄存器 13	X
REG12	[12]	常量布尔寄存器 12	X
REG11	[11]	常量布尔寄存器 11	X
REG10	[10]	常量布尔寄存器 10	X
REG9	[9]	常量布尔寄存器 9	X
REG8	[8]	常量布尔寄存器 8	X
REG7	[7]	常量布尔寄存器 7	X

REG6	[6]	常量布尔寄存器 6	X
REG5	[5]	常量布尔寄存器 5	X
REG4	[4]	常量布尔寄存器 4	X
REG3	[3]	常量布尔寄存器 3	X
REG2	[2]	常量布尔寄存器 2	X
REG1	[1]	常量布尔寄存器 1	X
REG0	[0]	常量布尔寄存器 0	X

8. 顶点着色器配置寄存器

全局寄存器包含多种配置和环境进行全局操作。

寄存器	地址	读/写	描述	复位值
FGVS_Config	0x7201C800	写	顶点着色器配置寄存器	0x00000000
FGVS_Status	0x7201C804	读	内部状态寄存器	0x00000000
FGVS_PCRange	0x72020000	读/写	顶点着色器程序开始和结束地址	0x01FF0000
FGVS_AttributeNum	0x72020004	读/写	当前文本属性的数量	0x00010001

FGVS_Config	位	描述	初始状态
Reserved	[31:1]	保留	0
ClrStatus	[1]	当此位被设置为 1 时, FGVS_Status 寄存器所有的值被清除。 清除操作完成后此位自动为 0.	0b
CopyPC	[0]	当此位设置为 1, FGVS_PCRange 寄存器值被复制到定点着色器 里面。复制完成后, 此位自动清除到 0。 在没有任何复制命令的时候, FGVS_PCRange 的值不被使用, 用 先前的值运行开始或结束地址。	0b

FGVS_PCRange	位	描述	初始状态
IgnorePCEnd	[31]	当此位设置为 1 时, PCEnd 值被忽略, 只有程序计数器栈空条	0

		件被程序终端应用。	
Reserved	[30:25]	保留	0
PCEnd	[24:16]	当程序计数器达到 PCEnd 的值，着色器程序在寄存器内的指令执行后被终止。这种方法可以保存指令槽和指令计数的数量用于执行。 另一种终止顶点着色器程序的方法： 顶点着色器程序可以通过额外的“ret”指令终止，“ret”指令使程序计数栈为空条件。通常“call”和“ret”指令成对工作。但是故意不匹配的“ret”使顶点着色器终止。通过这个特例，顶点着色器程序可以被终止。	0x1FF
Reserved	[15:9]	保留	0
PCStart	[8:0]	当顶点着色器开始运行，从指令存储器获取的第一个指令被储存在 PCStart 上。 这个寄存器值可以复制到顶点着色器程序计数器上，在着色器开始之前通过 PCCopy 寄存器完成。	0x00

FGVS_AttributeNum	位	描述	初始状态
Reserved	[31:4]	保留	0x0001000
InAttributeNum	[3:0]	顶点着色器输入属性的序号。 需要与当前着色器程序输入寄存器序号匹配。	0x1

9. 输入属性的索引寄存器

通常，顶点着色器输入寄存器的寄存器序号与主机输入属性的顺序相匹配。这种关系可以通过输入属性索引寄存器重新映射。主机的第 n 个输入属性实际上是位值的读数，通过索引寻找着色器内输入寄存器相应的寄存器序号可以指明。

寄存器	地址	读/写	描述	复位值
FGVS_InAttrIndex0	0x72020008	写	输入属性 0~3 的索引	0x03020100
FGVS_InAttrIndex1	0x7202000C	写	输入属性 4~7 的索引	0x07060504
FGVS_InAttrIndex2	0x72020010	写	输入属性 8~9 的索引	0x0B0A0908

FGVS_InAttrIndex0	位	描述	初始状态
Reserved	[31:28]	保留	0
Attrib3	[27:24]	输入属性 3 的索引	0x3
Reserved	[23:20]	保留	0
Attrib2	[19:16]	输入属性 2 的索引	0x2
Reserved	[15:12]	保留	0
Attrib1	[11:8]	输入属性 1 的索引	0x1
Reserved	[7:4]	保留	0
Attrib0	[3:0]	输入属性 0 的索引	0x0

FGVS_InAttrIndex1	位	描述	初始状态
Reserved	[31:28]	保留	0
Attrib7	[27:24]	输入属性 7 的索引	0x7
Reserved	[23:20]	保留	0
Attrib6	[19:16]	输入属性 6 的索引	0x6
Reserved	[15:12]	保留	0
Attrib5	[11:8]	输入属性 5 的索引	0x5
Reserved	[7:4]	保留	0
Attrib4	[3:0]	输入属性 4 的索引	0x4

FGVS_InAttrIndex2	位	描述	初始状态
Reserved	[31:12]	保留	0x0B0A0
Attrib9	[11:8]	输入属性 9 的索引	0x9
Reserved	[7:4]	保留	0
Attrib8	[3:0]	输入属性 8 的索引	0x8

10.输出属性的索引寄存器

通常，像素着色器输入寄存器的寄存器序号与顶点着色器的输出寄存器的序号相对应。这种关系可以通过输出属性索引寄存器重新映射。第 n 个输出属性实际上是位值的写入，通过索引寻找着色器内输出寄存器相应的寄存器序号可以指明。输入寄存器 0 从输出寄存器 1 获得数值，因为顶点着色器的输出寄存器 0 的位置是特别指明的。因此，顶点着色器的输出寄存器 1-8 与像素着色器的 0-7 相对应。

寄存器	地址	读/写	描述	复位值
FGVS_OutAttrIndex0	0x72020014	写	输出属性 0~3 的索引	0x03020100
FGVS_OutAttrIndex1	0x72020018	写	输出属性 4~7 的索引	0x07060504
FGVS_OutAttrIndex2	0x7202001C	写	输出属性 8~9 的索引	0x0B0A0908

FGVS_OutAttrIndex0	位	描述	初始状态
Reserved	[31:28]	保留	0
Attrib3	[27:24]	输出属性 3 的索引	0x3
Reserved	[23:20]	保留	0
Attrib2	[19:16]	输出属性 2 的索引	0x2
Reserved	[15:12]	保留	0
Attrib1	[11:8]	输出属性 1 的索引	0x1
Reserved	[7:4]	保留	0
Attrib0	[3:0]	输出属性 0 的索引	0x0

FGVS_OutAttrIndex1	位	描述	初始状态
Reserved	[31:28]	保留	0
Attrib7	[27:24]	输出属性 7 的索引	0x7
Reserved	[23:20]	保留	0
Attrib6	[19:16]	输出属性 6 的索引	0x6
Reserved	[15:12]	保留	0
Attrib5	[11:8]	输出属性 5 的索引	0x5
Reserved	[7:4]	保留	0

Attrib4	[3:0]	输出属性 4 的索引	0x4
---------	-------	------------	-----

FGVS_OutAttrIndex2	位	描述	初始状态
Reserved	[31:12]	保留	0x0B0A0
Attrib9	[11:8]	输出属性 9 的索引	0x9
Reserved	[7:4]	保留	0
Attrib8	[3:0]	输出属性 8 的索引	0x8

42.5 图元引擎

42.5.1.概述

图元引擎是一个有线硬件模块，可以处理一系列的操作，包括原始集合，平面着色，视图体裁剪，视图划分和视图映射。视图体裁剪划分为近/远平面裁剪操作和左/右/顶/底平面裁剪操作。在图元引擎内，只处理近/远平面裁剪操作，除了当裁剪的顶点 w -坐标为 0 以外。使用窗口裁剪操作，其它裁剪操作可以通过三角设置引擎和光栅化引擎处理。当顶点的 W -坐标为 0 时，图元引擎运行其他裁剪操作避免输出顶点的 W -坐标变为 0。

图 42-9 所示图形表示出图元引擎的概念功能级模块图，解释说明了图元引擎的操作。

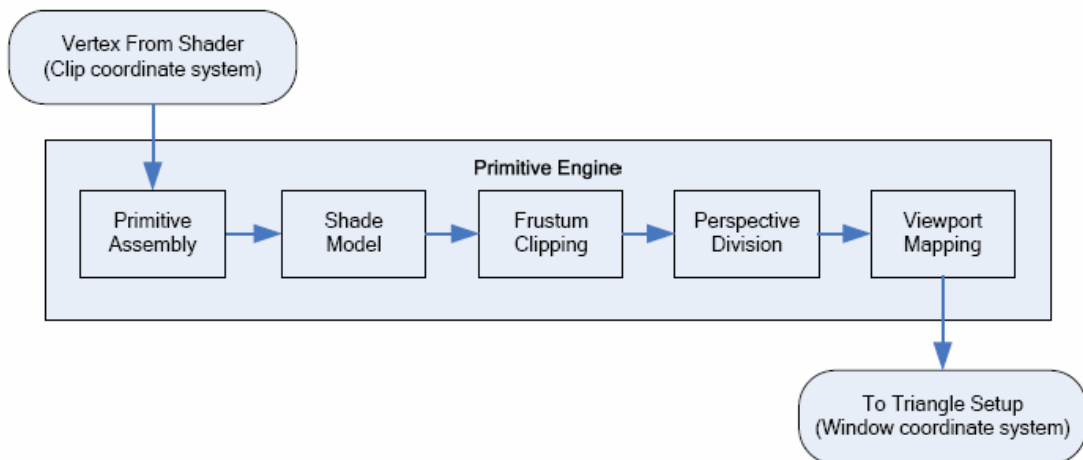


图 42-9 图元引擎的概念功能级模块图

42.5.2.图元引擎特殊寄存器

在图元引擎内有两种特殊寄存器。一个寄存器用于表示顶点信息，如图元类型，相关数据的数目以及色调模式。另一个寄存器用于表示视口参数。

42.5.3.顶点背景寄存器

在图元引擎内，顶点背景寄存器将一个输入顶点分为 3 种信息进行处理。29-29 之间的位区域表示返回的图元类型。支持图元引擎寄存器，三角形，三角扇形，三角地带，线，线环，带，点和点带。其他类型的如多边形，方形，方形带不支持，但是对于这些形状的位已经保留了。每个图元类型的位区域都相互正交。这就暗示着当用点带模式是，点带的位必须从新设置。

13-10 之间的位区域表示编码模式下与顶点相关数据的序号。MSB 的位是 13，LSB 的位是 10.相关数据的序号位宽度假设为 128。如果顶点相关数据是 5 个 32 位值，那么相关数据的序号必须设置为 2。当想要使用顶点着色器点尺寸模式时，相关数据的序号必须加 1。如上所述，相关数据的序号与顶点着色器的变量和顶点着色器程序点尺寸模式有关，因此相关数据需要的范围是 0-9。可以通过下面的式子表示相关数据的序号和 VSOUT:

$$VSOUT = \frac{\text{变量数}}{4} + 1, \quad \text{顶点程序点尺寸模式}$$

$$VSOUT = \frac{\text{变量数}}{4}, \quad \text{其他情况}$$

9-0 之间的位区域表示色调模式信息。位 9 是主标志，表示色调模式是平滑的或平坦的。0-8 位区域与顶点着色器输出有关，为了用顶点着色器输出 Slot0 为平坦颜色通道，位 9 和位 4 必须设置为 1.

位 31 和 30 是内部使用的位，不要触摸这两个位。

注意：在 3D 图形内，顶点着色器支持的输入属性数量可达 8 个，变量数可达 32 个。

寄存器	地址	读/写	描述	复位值
FGVS_VERTEX_CONTEXT	0x72030000	读/写	顶点背景格式定义	0x00000000

FGVS_VERTEX_CONTEXT	位	描述	初始状态
INTUSE	[31:30]	保留的位，用于内部使用。不要触摸这些位。	0b
POLYGON	[29]	保留的位，用于多边形图元类型	0b
QUADS	[28]	保留的位，用于方形图元类型	0b
QUADSTRIP	[27]	保留的位，用于方形带图元类型	0b
TRIANGLES	[26]	1b=三角形图元类型 0b=非三角形图元类型	0b
TRIANGLEFAN	[25]	1b=三角扇形图元类型 0b=非三角扇形图元类型	0b
TRIANGLESTRIP	[24]	1b=三角带形图元类型 0b=非三角带形图元类型	0b
LINES	[23]	1b=线形图元类型 0b=非线形图元类型	0b
LINELOOP	[22]	1b=线环形图元类型 0b=非线环形图元类型	0b
LINESTRIP	[21]	1b=线带形图元类型 0b=非线带形图元类型	0b
POINTS	[20]	1b=点图元类型 0b=非点图元类型	0b
POINTSPRITE	[19]	1b=点带图元类型 0b=点带角形图元类型	0b
VERTEXPROGRAMPOINTSIZ	[18]	1b=顶点着色器程序点尺寸模式 0b=非顶点着色器程序点尺寸模式	0b
Reserved	[17:14]	保留	0
VSOUT[3:0]	[13:10]	顶点着色器使用的位置，输出数目	0x0

FLAT_SHADE	[9]	1b=使用平坦色调模式 0b=使用平滑色调模式	0b
FLAT_MODEL8	[8]	1b=顶点着色器输出 8 使用平坦色调模式 0b=顶点着色器输出 8 使用平滑色调模式	0b
FLAT_MODEL7	[7]	1b=顶点着色器输出 7 使用平坦色调模式 0b=顶点着色器输出 7 使用平滑色调模式	0b
FLAT_MODEL6	[6]	1b=顶点着色器输出 6 使用平坦色调模式 0b=顶点着色器输出 6 使用平滑色调模式	0b
FLAT_MODEL5	[5]	1b=顶点着色器输出 5 使用平坦色调模式 0b=顶点着色器输出 5 使用平滑色调模式	0b
FLAT_MODEL4	[4]	1b=顶点着色器输出 4 使用平坦色调模式 0b=顶点着色器输出 4 使用平滑色调模式	0b
FLAT_MODEL3	[3]	1b=顶点着色器输出 3 使用平坦色调模式 0b=顶点着色器输出 3 使用平滑色调模式	0b
FLAT_MODEL2	[2]	1b=顶点着色器输出 2 使用平坦色调模式 0b=顶点着色器输出 2 使用平滑色调模式	0b
FLAT_MODEL1	[1]	1b=顶点着色器输出 1 使用平坦色调模式 0b=顶点着色器输出 1 使用平滑色调模式	0b
FLAT_MODEL0	[0]	1b=顶点着色器输出 0 使用平坦色调模式 0b=顶点着色器输出 0 使用平滑色调模式	0b

42.5.4. 视口参数寄存器

视口转变由适口宽度和高度像素 P_x 和 P_y 决定，视口中心 (O_x, O_y) 也是像素模式。顶点窗口坐标

$$\begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix}$$

由下面的式子表示：

$$\begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix} = \begin{pmatrix} (p_x/2)x_d + o_x \\ (p_y/2)y_d + o_y \\ [(f-n)/2]z_d + (n+f)/2 \end{pmatrix}$$

应用到 Z_d 的因子和补偿区通过近深度范围 n 和远深度范围 f 编码。视口中心 (O_x, O_y) 可以用 $(x_0 + p_x/2, y_0 + p_y/2)$ 表示，假设视口原点为 (x_0, y_0) 。在 3D 图形内，使用 Y 轴滚动窗口坐标系统。为了产生 y 周滚动窗口坐标系统，可以简单的替换 y 轴相关等式， $y_w = (p_y/2)y_d + o_y$ 和 $o_y = y_0 + p_y/2$ ， $y_w = (-p_y/2)y_d + o_y$ 和 $o_y = (\text{窗口高度}) - y_0 - p_y/2$ 。

寄存器	地址	读/写	描述	复位值
FGPE_VEIWPOROT_OX	0x72030004	读/写	视口中心 X 坐标	0xX
FGPE_VEIWPOROT_OY	0x72030008	读/写	视口中心 Y 坐标	0xX
FGPE_VEIWPOROT_HALF_PX	0x7203000C	读/写	视口宽度的一半	0xX
FGPE_VEIWPOROT_HALF_PY	0x72030010	读/写	视口高度的一半	0xX
FGPE_DEPTHRANGE_HALF_F_SUB_N	0x72030014	读/写	远深度范围的一般减近深度范围的一半	0xX
FGPE_DEPTHRANGE_HALF_F_ADD_N	0x72030018	读/写	远深度范围的一半加近深度范围的一半	0xX

P_x : 视口宽度像素

P_y : 视口高度像素

X_0 : 窗口坐标系统内视口中心的 x 坐标

Y_0 : 窗口坐标系统内视口中心的 y 坐标

n : 深度范围的近值

f : 深度范围的远值

H : 窗口高度像素

	位	描述	初始状态
FGPE_VIEW PORT_OX	[31:0]	视口中心的 x 坐标。 $x_0 + \frac{p_x}{2}$	0xX