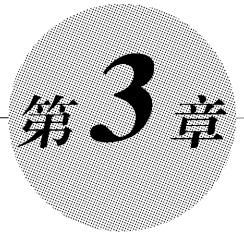


Geode LX 处理器的工作方式

Geode LX 处理器有两种主要的工作方式：实地址方式和保护虚地址方式。实地址方式是为了与 8086 兼容而设置的方式。在实地址方式下，具有 32 条地址线的 Geode LX 处理器只有低 20 条地址线起作用，能寻址 1MB 的物理地址；此时，Geode LX 处理器相当于一个快速的 8086，虽然可以使用 32 位的数据寄存器，但远不能充分发挥 Geode LX 处理器的全部功能。保护虚地址方式是 Geode LX 处理器的主要工作方式，在此方式下，全部 32 条地址线都能寻址，故可寻址高达 4GB 的物理存储器；在保护方式下，Geode LX 处理器支持虚拟存储器的功能，一个任务可运行多达 16K 个段，每个段最大可为 4GB；在保护方式下运行的程序分为 4 个特权等级：0、1、2 和 3，操作系统核心运行在最高特权等级 0，用户程序运行在最低特权等级 3。Geode LX 处理器中有完善的特权检查机制，既能实现资源共享又能保证程序和数据的安全和保密以及任务之间的隔离。在保护方式下，Geode LX 处理器支持多用户多任务操作系统，可以用一条指令实现任务切换，而且任务的环境得到了很好的保护，Geode LX 处理器的芯片内包含一个存储管理单元 MMU，在保护方式下可以实现分页，通过两级页表，可以把物理地址映射到线性地址空间的任何区域。总之，在保护虚地址方式下，Geode LX 处理器具有很强的功能，保护方式是 Geode LX 处理器的主要工作方式。而且，Geode LX 处理器在保护虚地址方式下，有一种虚拟 8086 方式，可以在多任务的条件下，使有的任务运行 MS-DOS，这是一种与 8086 兼容但又不同于实地址方式的工作方式。

除上述主要工作模式外，Geode LX 处理器还有一种主要用于系统管理的模式，即系统管理模式(SMM)，这是一种为系统控制活动运行设计的模式，它对于常规的系统软件是透明的。电源管理是系统管理模式的一种流行使用。SMM 的最初目标由基本输入-输出系统(BIOS)和特殊的低级设备驱动程序使用。SMM 的代码和数据存储在 SMM 内存区，它由 SMM 输出信号与主内存隔离。

SMM 由系统管理中断(SMI)进入。直至识别 SMM，处理器进入和切换至不同的 SMM 地址空间，在那儿处理程序存放和执行。在 SMM 中，处理器支持实模式寻址，具有 4GB 段界限和 16 位的默认的操作数、地址和堆栈尺寸(能用前缀超越这些默认值)。

3.1 实地址方式

实地址方式和保护虚地址方式的区分是由控制寄存器 CR0 的最低位 PE 位决定的。若 PE 位为 0，则工作在实地址方式；若 PE＝1，则工作在保护虚地址方式。

Geode LX 处理器在系统复位后，CR0 的 PE＝0，即工作在实地址方式。在经过了必要的初始化以后（后面将会详细介绍），用 MOV 指令使 CR0 加载一个 PE 位等于 1 的新的操作数，就使工作方式切换到保护虚地址方式。

在实地址方式下的存储器寻址与 8086 是一样的，32 位地址线中的 A31～A20 不起作用。由段寄存器（CS、SS、DS 和 ES）的内容乘以 16 作为段基地址，加上 16 位的段内偏移量形成 20 位的物理地址。在实地址方式下，每一个段最大可达 64KB。所有的段都是可读、写和执行的。在实地址方式下的内存是不能分页的，故线性地址和物理地址是统一的。

在实地址方式下运行的程序不分特权等级。实际上，实地址方式下的程序相当于工作在特权级 0，它能执行控制寄存器（CR0 和 CR3）传送指令，加载 GDTR、LDTR 和 TR 等特权指令。除保护虚地址方式下的一些专用指令之外，所有其他指令都能在实地址方式下执行。所以，系统复位以后，要在实地址方式下，初始化 gdt、idt 和两级页表，加载 CR3，然后通过加载 CR0 使 PE ＝1，才能进入保护虚地址方式。

实地址方式下不能实现多任务，所以，Geode LX 处理器的实地址方式，是系统复位后向保护虚地址方式过渡的一种方式。

一部分 Geode LX 处理器在 DOS 支持下工作，只工作在实地址方式，这主要是为了与 8086 兼容，能运行 DOS 支持下的软件。

Geode LX 处理器在实地址方式下有两个内存保留区：系统初始化区和中断向量表区。

Geode LX 处理器在复位以后，CS 寄存器的值为 F000H，而 IP 被初始化为 FFF0H，而且系统强迫地址总线的高 12 位为 1。所以，初始化后的入口地址为 FFFFFFF0H，从 FFFFFFF0H～FFFFFFFFH 为初始化的保留区，通常在 FFFFFFF0H 处存放一条段间跳转指令，转至系统的入口处。

实地址方式与 8086 方式相似，在内存的 00000000H 至 000003FFH 的 1KB 区域内，存放一个具有 256 个向量的中断向量表，每一向量对应着一个 4 个字节的中断服务程序的入口地址（两个字节的段寄存器值，两个字节的段内偏移量）。

在实地址方式下的中断与异常和保护虚地址方式下有较大的区别。这主要涉及系统程序员的工作，本书中不作分析。

3.2 保护虚拟地址方式

3.2.1 保护方式下的寻址机制

在保护方式下，一个存储单元的地址也是由段基地址和段内偏移量两部分组成的。从段内偏移量来说，除能扩展到全地址（32 位）外与实地址方式下区别不大，寻址方式的

根本区别在于如何确定段基地址。在实地址方式下，段寄存器的内容乘以16（即左移四位）就形成段基地址，故段基地址是20位的，只能寻址1MB；在保护方式下，段基地址也是32位的，所以就不能由段寄存器的内容直接形成32位的段基地址，而要经过转换。于是在内存中就有一个表，每一个内存段对应着表中的一项，此项中包含32位的段基地址。为了适应多用户、多任务操作系统的需要，一个段还要有一些其他信息，例如，段的大小（界限）和段的一些读写权限、段的类型等。在Geode LX处理器中，一个段用一个8字节的描述符来描述，这些描述符构成的表，称为描述符表。

由描述符中所规定的段基地址再加上32位的段内偏移量就可以寻址一个存储单元，如图3-1所示。

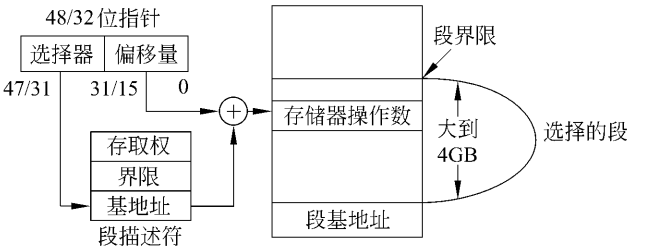


图3-1 保护方式下的寻址

由段基地址（32位）和段内偏移量（32位）形成的地址称为线性地址（32位）。在Geode LX处理器片内有分页的MMU，当启用分页机制时（CR0的最高位PG＝1），通过分页机制可以把线性地址转换为存储器的物理地址，如图3-2所示。

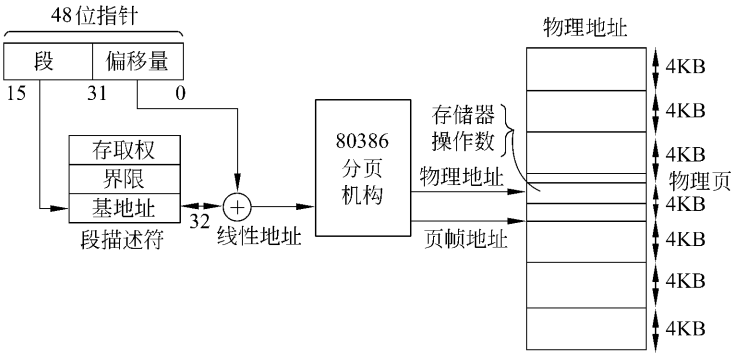


图3-2 分页和分段

当不启用分页机制时（CR0的最高位PG＝0），线性地址即为内存的物理地址。

保护方式下的段内偏移量为32位，故一个段最大可达4GB。

3.2.2 全局描述符表和局部描述符表

在Geode LX处理器中，有三种类型描述符表：全局描述符表（gdt）、局部描述符表（ldt）和中断描述符表（idt）。在整个系统中，全局描述符表和中断描述符表都只有一个，局部描述符表可以有若干个，每一个任务一个。

每个描述符表本身形成一个段，最多可以有 8K(8192)个描述符。但 Geode LX 处理器中，最多只能处理 256 个中断向量，故中断描述符表最多只包含 256 个中断描述符。每个描述符表构成一个段，也有段的基地址、段的界限和其他特性，即也有一个相应的描述符来描述。这样的描述符必须放在全局描述符表中。

1. 全局描述符表(gdt)

在全局描述符表中，包含着系统中每一个任务都可能(或可以)访问的段的描述符，通常包含操作系统使用的码段、数据段和堆栈段，各种任务状态段，系统中所有的 ldt 表的描述符等。

2. 局部描述符表(ldt)

通常，操作系统的设计者使每一个任务都有自己的 ldt。ldt 既包含了此任务所使用的码段、数据段、堆栈段描述符；也可包含此任务所使用的一些控制描述符，如任务门、调用门描述符。

使用 ldt 这样的数据结构，就可以使指定任务的码段、数据段等与别的任务相隔离以达到保护。

使用 gdt 和 ldt 这两种数据结构就可以达到既保护又可共享全局数据的目的。

从系统的虚拟地址空间来看，整个虚拟地址空间可以分成两部分，一部分空间的描述符在全局描述表中，另一部分空间的描述符在局部描述符表中。每一个表都可以包含多达 8192 个描述符(即对应的空间可由 8192 个段组成)，每一个段最大可为 4GB。当切换任务时，ldt 就切换为新任务的 ldt，而 gdt 是不变的。因此，由 gdt 所映像的虚拟地址空间对所有的任务是公共的；而 ldt 所映像的虚拟地址空间，只局限于任务，并随着任务而改变。

对于一个系统来说，操作系统是面向所有任务的，它应该在 gdt 的映像中。一些全局性的数据、表格以及公用的实用程序等也应在 gdt 的映像中。

上述的全局和局部地址空间的情况如图 3-3 所示。它既可做到互相隔离和保护，也

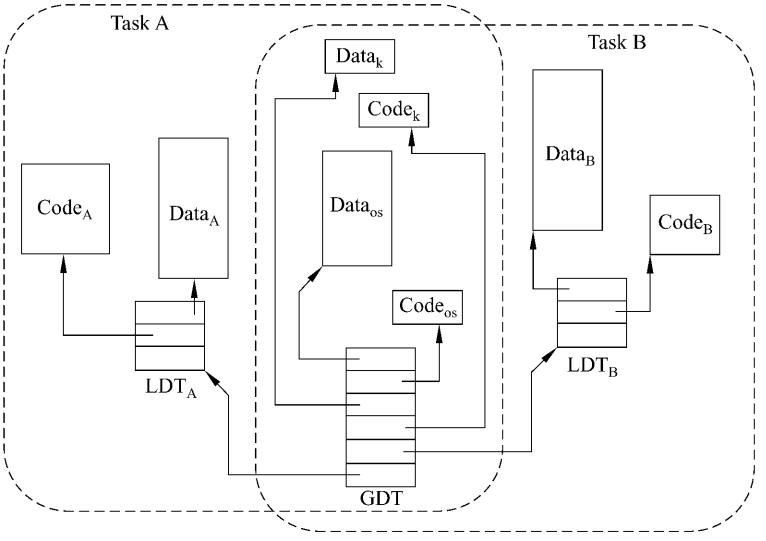


图 3-3　全局和局部地址空间

可以做到全局数据的共享。

3.2.3　描述符

在保护虚地址方式下的每一个段，都有一个相应的描述符。描述符由 8 个字节组成，它包含了此段的基地址(32 位)、段的大小(20 位)以及段的类型等一些主要特性。

在 Geode LX 处理器中，主要有以下两种类型描述符。

- 段码和数据段描述符。
- 特种数据段和控制描述符。

后一种又分为特种数据段描述符和控制(门)描述符两大类。

1. 码段和数据段描述符

其一般格式如图 3-4 所示。图中规定了 32 位的段基址(由基址 31…24、基址 23…16 和基址 15…0 三部分构成)、20 位段界限(由界限 19…16 和界限 15…0 两部分构成)。另有一粒度位 G。若 G= 0，段长度以字节为粒度，20 位段界限可定义段的大小为 1 兆字节(1MB)；若 G=1，段长度以页为粒度，每页为 4KB，段的界限为 1M 页，可定义段的大小为 1M×4KB=4GB。

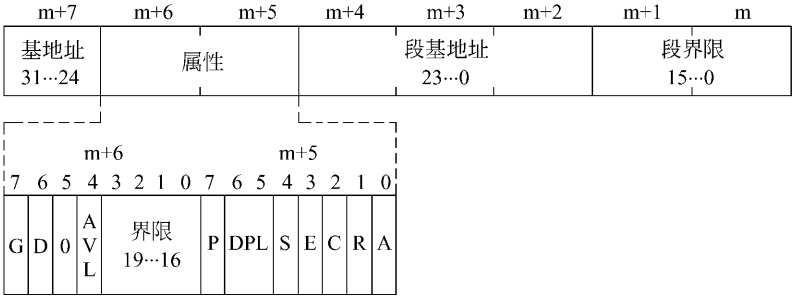


图 3-4　码段和数据段描述符

描述符中有一个字节，称为段的访问权字节，它描述了段的一些重要特性。

(1) 其中的高 4 位在所有的段描述符中都是相同的。

最高位 P(Present)为存在位，若 P=1，此段被映像到物理存储器；若 P=0，无物理存储器映像存在。

第 6 和第 5 位 DPL(Descriptor Privilege Level)为描述符特权级，规定了此段的特权级，用于特权检查，以决定能否对此段进行访问。

第 4 位 S(Segment)区分两大类描述符，S=0 时，为特殊数据段或控制描述符；S=1 时，为码段或数据段描述符。

(2) 访问权字节的低 4 位，在 S=0 和 S=1 时是不同的。在 S=1 即为码段或数据段描述符时，其作用如表 3-1 所示。它反映了码段或数据段以及段能否被读写访问。

表 3-1 码段和数据段的访问权字节

位	命　名	功　能
7	存在(P)	若 P=1,段映像到物理存储器; 若 P=0,无物理存储器映像存在,描述符无效
6,5	描述符特权级(DPL)	段的特权属性,用于访问时的特权测试
4	段描述符(S)	若 S=1,码或数据(包括堆栈)段描述符; 若 S=0,特种数据段或控制(门)描述符
在 E=0 的情况下: 3 2 1	可执行(E) 扩展方向(ED) 可写(W)	若 E=0,不可执行,为数据段描述符。 若 ED=0,向上扩展,偏移量必须≤界限; 若 ED=1,向下扩展,偏移量必须>界限。 若 W=0,数据段不能写入; 若 W=1,数据段可写入。 如果为数据段,必须: S=1 E=0
在 E=1 的情况下: 3 2 1	可执行(E) 一致(C) 可读(R)	若 E=1,可执行,为码段描述符。 若 C=1,当 CPL≥DPL 并且 CPL 保持不变时,码段只能执行。 若 R=0,码段不可读; 若 R=1,码段可读。 如果为码段,必须: S=1 E=1
0	访问(A)	若 A=0,段尚未被访问; 若 A=1,段已被访问

段的访问权字节是段的极其重要的属性。描述符的各个字段,可以由相应的指令来读取和设置。

在描述符中,比访问权字节地址高的一个字节的有些位也是很重要的,其中的 G 位如上所述是界限的粒度位。D 位也很重要,其作用如下:

(1) 码段描述符的 D 位用于设置由指令所引用的地址和操作数的默认值。若 D=1,指示默认值是 32 位地址、32 位或 8 位操作数,这是 Geode LX 处理器在保护方式下的正常设置;若 D=0,指示默认值是 16 位地址、16 位或 8 位操作数,这在 Geode LX 处理器中是为了执行 80286 的程序而设置的。

由 D 位所设置的默认值,可以通过地址前缀和操作数前缀加以改变。

(2) 对于设置为向下扩展的段,D 位决定段的上边界。若 D=1,则指示段的上边界为 4GB;若 D=0,则指示段的上边界为 64KB。

(3) 由 SS 寻址的段,若 D 位=1,规定用 ESP 作为指针,堆栈操作是 32 位的;若 D=0,则用 SP 作为指针,堆栈操作是 16 位的。

2. 特种数据段和控制描述符

此类描述符的一般格式如图 3-5 所示。

其中,TYPE 字段共 4 位,用此 4 位二进制的值区分描述符的具体类型。在 Geode LX 处理器中定义如下:

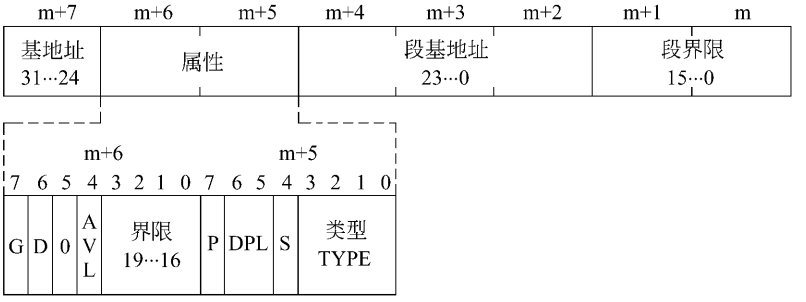


图 3-5 特种数据段和控制描述符格式

TYPE	类 型	TYPE	类 型
0	未定义	8	未定义
1	有效的 286 TSS	9	有效的 386 TSS
2	ldt 描述符	A	未定义
3	286 TSS 忙	B	386 TSS 忙
4	286 调用门	C	386 调用门
5	任务门(对 286 和 386 都适用)	D	未定义
6	286 中断门	E	386 中断门
7	286 陷阱门	F	386 陷阱门

3. 特种数据段描述符

x86 系统中有两种特种数据段,即局部描述符表的描述符和任务状态段 TSS(Task Status Segment)的描述符。

ldt 中包含了一个任务所要访问的段的所有描述符,但此表本身构成一个存储器数据段,它也有段基地址、段的界限和段的属性,需要一个描述符与之相对应,这样的描述符当然要放在 gdt 中。

当切换任务时,一个任务的环境(状态)需要保存(离去的任务)或需要恢复(进入的任务)。这些环境是保存在内存的某个段内的,如图 3-6 所示,称为任务状态段(TSS)。TSS 是一个段,它具有与一般段同样的特性,需要有一个描述符与之对应。当然,这样的描述符也存放在 gdt 中。

任务状态段描述符分为 286 和 386 两种,每种都还有可用或忙两种情况,故共有 4 种。ldt 描述符和 4 种 TSS 描述符的格式如图 3-5 所示,由于只是 TYPE 字段的值不同,故不再赘述。

4. 控制(门)描述符

程序在运行中会发生转移,系统会进行任务切换,外部事件会引起中断,指令的执行会引起异常,总之,会发生控制转移。转移至何处? 转移可能在同一段内进行,但更普遍的是段间转移(任务切换、中断和异常,一定都是段间转移;程序的跳转与调用也可能是段间的)。如何确定目标段和相应的入口呢? 控制转移通常会涉及特权级的变换,所以 x86 中设置了调用门、任务门、中断门和陷阱门。调用门用于在程序中调用子程序、过程和函

31		0	
I/O Permission Bitmap Offset	0000000000000000	T	64h
0000000000000000	LDT		60h
0000000000000000	GS		5ch
0000000000000000	FS		58h
0000000000000000	DS		54h
0000000000000000	SS		50h
0000000000000000	CS		4ch
0000000000000000	ES		48h
EDI			44h
ESI			40h
EBP			3ch
ESP			38h
EBX			34h
EDX			30h
ECX			2ch
EAX			28h
EFLAGS			24h
EIP			20h
CR3			1ch
0000000000000000	SS2		18h
ESP2			14h
0000000000000000	SS1		10h
ESP1			0ch
0000000000000000	SS0		8
ESP0			4
0000000000000000	LINK		0

图 3-6　任务状态段

数；任务门用于任务切换；中断门用于外部事件引起的中断；陷阱门用于异常处理。

每一种门都要涉及转移的目标段及在段内的入口，所以门描述符与段描述符非常相似。其一般格式如图 3-7 所示。其中包括目标段的选择子（通过选择子来确定相应的段），入口点的偏移量（32 位）以及有关的特性。TYPE 字段的值区分了几种不同的门描述符。只有在调用门中才有 DC 字段（在其他类型的门中，此字段为 0），它规定了从调用

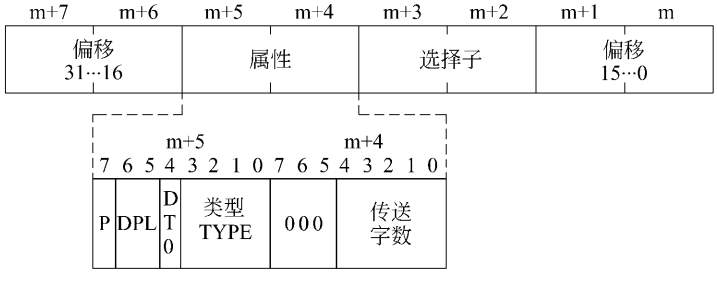


图 3-7　门描述符

者的堆栈中复制多少个双字到被调用者的堆栈中，用于传递参数。在任务门中，一般不用入口点的偏移量。

3.2.4 选择子

每一个段相应的描述符在 gdt 或 ldt 中。要选择目标段，就要从 gdt 或 ldt 中取出相应的描述符，而目标段是由段寄存器规定的。所以，在保护虚地址方式下，段寄存器的内容就成为段选择子，由它从 gdt 或 ldt 中读取对应的描述符。选择子的格式如图 3-8 所示。

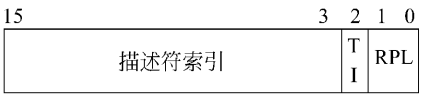


图 3-8 选择子

图 3-8 中，最低两位为请求特权级 RPL (Requested Privlege Level)，也用于段访问时的特权测试；第 2 位 TI(Table Indicator)指示从哪个描述符表中去读取描述符。若 TI=0，指示从 gdt 中读取描述符；若 TI=1，指示从 ldt 中读取描述符。剩下的 13 位即为此段在描述符表中的索引，$2^{13}=8192$，故可区分 8192 个描述符。这就是一个描述符表最大能包含 8192 个描述符的原因。

有一个特殊的选择子称为空(Null)选择子，它的 DI(索引)字段为 0，TI 字段也为 0，而 RPL 字段可为任意值。空选择子有特定的用途，当用空选择子进行存储器访问时会引起异常。

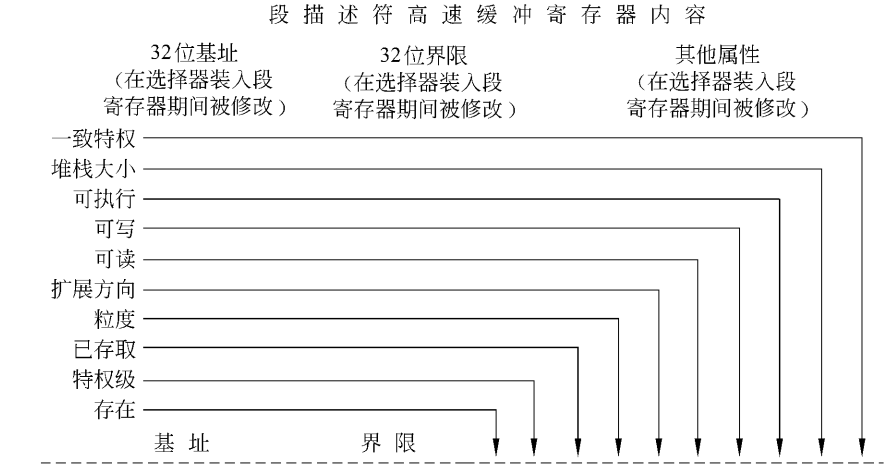
空选择子是特别定义的，它不对应于 gdt 中的第 0 个描述符，所以，gdt 中的第 0 个描述符是不用的，必须置为 0。一个选择子，当它的 DI(索引)字段为 0 而 TI 字段为 1 时，就不是空选择子，而是选择 ldt 中的第 0 个描述符。

3.2.5 段描述符的高速缓冲寄存器

Geode LX 处理器中的每一个段，都有一个段寄存器用以装载段选择子，用于从描述符表中取出此段的描述符。这样，在保护虚地址方式下，要访问一个存储单元，首先要以相应的段寄存器(码段为 CS，数据段为 DS、ES、FS 或 GS，堆栈段为 SS)作为选择子，从相应的描述符表(gdt 或 ldt)中取出描述符(一次存储器访问)，找到此段的基地址，与段内偏移量相加得到存储单元的线性地址(若不考虑分页机制则为物理地址)，再进行一次存储器访问才能取出所需的指令或数据。这样，每访问一个存储单元，需要进行两次存储器访问操作，就会大大降低运行速度。为了消除每次存储单元的访问，都要先取出描述符。Geode LX 处理器在硬件上增加了一个用户不可见的段描述符——高速缓冲寄存器(Cache)。每一个段寄存器都有一个对应的高速缓冲寄存器(或称为段寄存器的隐藏部分)。每当用一个选择子加载一个段寄存器时，Geode LX 处理器的硬件自动从描述符表中取出相应的描述符，加载至相应的高速缓冲寄存器中。一旦装入，此后对此段的访问都使用此高速缓冲寄存器中的描述符信息，而不用再去取描述符，直至对段寄存器重新装载。

Geode LX 处理器在保护虚地址方式下的段高速缓冲寄存器如图 3-9 所示。其中包括 32 位的段基地址，32 位已经转换为字节粒度的段界限以及 10 个特性位。

在实地址方式下，每个段寄存器也有相应的高速缓冲寄存器，只是内容上与保护虚地



CS	每个段描述符基址	每个段描述符界限	p	d	d	d	d	d	N	Y	—	d
SS	每个段描述符基址	每个段描述符界限	p	d	d	d	d	r	w	N	d	—
DS	每个段描述符基址	每个段描述符界限	p	d	d	d	d	d	d	N	—	—
ES	每个段描述符基址	每个段描述符界限	p	d	d	d	d	d	d	N	—	—
FS	每个段描述符基址	每个段描述符界限	p	d	d	d	d	d	d	N	—	—
GS	每个段描述符基址	每个段描述符界限	p	d	d	d	d	d	d	N	—	—

关键字：Y=是
N=否
d=每段描述符
p=每段描述符，描述符必须表示“存在”以避免异常11(在SS情况下为异常12)
r=每段描述符，但描述符必须表示“可读”以避免异常13(SS的特殊情况)
w=每段描述符，但描述符必须表示“可写”以避免异常13(SS的特殊情况)
—=不用于该段高速缓冲寄存器

图 3-9　段高速缓冲寄存器

址方式下有所不同，如图 3-10 所示。

例如，段基地址仍是 32 位，但其值为 16×相应的段寄存器值(实际有效的为低 20 位，只能寻址 1MB)。32 位的段界限每个段都固定为 0000FFFFH，即为 64KB。10 个特性位中的许多位也是固定的，如存在位始终是 Y(即存在)，特权级始终为 0(实地址方式，相当于工作在 0 特权级)等。

3.2.6　Geode LX 处理器中的特权级

Geode LX 处理器中的每一个程序都是在一定的特权等级下工作的。为了支持多用户多任务操作系统，使操作系统程序和用户的任务程序分离，任务和任务分离，在 Geode LX 处理器中提供了 4 个特权等级。利用这个特权系统，可控制特权指令和 I/O 指令的使用，并控制对段和段描述符的访问。

这种 4 级的特权系统如图 3-11 所示。

这实际上是在小型计算机以上的系统中采用的用户/管理员特权方式的扩展，而这种用户/管理员方式也是 Geode LX 处理器的分页机制所支持的。特权级的编号为 0 到 3，0 是 最高特权级，3是最低特权级。在一个任务中的特权级是用来提供保护的(任务之间的