上海大学

SHANGHAI UNIVERSITY

操作系统（一）研讨报告

组　　号　　第 7 组

研 讨 日 期　　2022年9月23日

学号	姓名	分工
20121034	胡才郁	基本概念了解与核心要点讨论，报告撰写
20121706	张俊雄	基本概念了解与核心要点讨论，报告撰写
20122191	胡天磊	基本概念了解与核心要点讨论，保护模式演讲，报告撰写
20122264	胡峻豪	基本概念了解与核心要点讨论，PPT 制作，实模式演讲，报告撰写

一、研讨选题

描述 x86 中实模式和保护模式的寻址区别，并说明在这两种模式中的逻辑地址、线性地址、和物理地址的关系。

二、要点小结

1、x86中的实模式寻址方式

1.1 实模式概述

实模式，也称为实地址模式（real address mode），是所有x86兼容CPU下的一种操作模式。实模式的特点是20 bit分段内存地址空间（精确到1 MB的可寻址内存）以及对所有可寻址内存，I/O地址和外设硬件的无限制直接软件访问。实模式不支持内存保护，多任务处理或代码权限级别。

最早期的8086 CPU只有一种工作方式，那就是实模式，而且数据总线为16位地址总线为20位，实模式下所有寄存器都是16位。而从80286开始就有了保护模式从80386开始CPU数据总线和地址总线均为32位，而且寄存器都是32位。80386以及现在的奔腾，酷睿等等CPU为了向前兼容都保留了实模式，有两种做法可以采纳：

①x86 CPU在重置（reset）时都以实模式启动；

②x86 CPU以其他模式启动，从做系统起来后再模拟（emulate ）实模式。目前，几乎所有x86 CPU都采用第一种方式。

1.2 实模式寻址方式

实模式的内存寻址方式：分段寻址（段基址*16+段内偏移）。

因为在16位CPU中寄存器只有16位，及最大只可以寻址216=64K 的空间，然而CPU的地址线却有20根，即最大可支持220=1M的地址空间，所以为了能够让CPU利用更大的内存空间，就需要采用分段寻址的方式，即把1M的内存分为许多段（最大为64K）,在段基址寄存器中存放段基地址，而在段内偏移寄存器中存放段内偏移量，在寻址时，就把CS<<4+IP，即可得到实际内存地址。

CPU要访问内存（从内存上取指令或数据），需要向总线上发送要访问的内存地址，在实模式下计算地址：

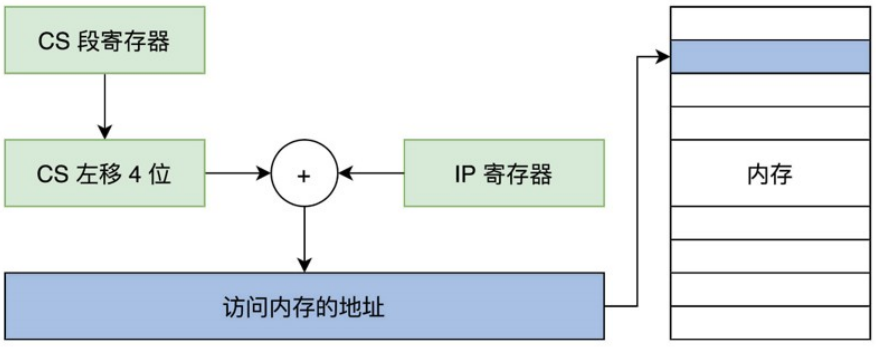

图1. 指令地址寻址

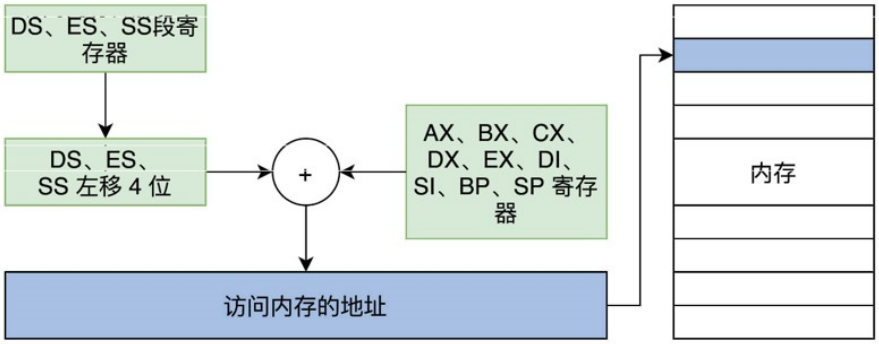

图2. 数据地址寻址

对于指令地址，CPU会通过CS和IP寄存器的值组合而来，值为CS所存储的地址左移4位加IP的值：（CS << 4） + IP。

对于数据地址，CPU会通过DS,ES,SS加上AX,BX,CX,DX,EX,DI,SI,BP,SP寄存器组合而来,DS一般用来存放数据段内容比如C语言全局变量；SS则是栈的基地址，SS搭配SP来使用，一般用来访问C语言临时变量和函数栈信息。地址计算规则和指令地址计算规则类似。

2、x86中保护模式寻址方式

保护模式中，段模式下是利用一个称作段选择子的偏移量，从而到描述符表找到需要的段描述符，而这个段描述符中就存放着真正的段的物理首地址，再加上偏移量，就找到真正的物理地址。

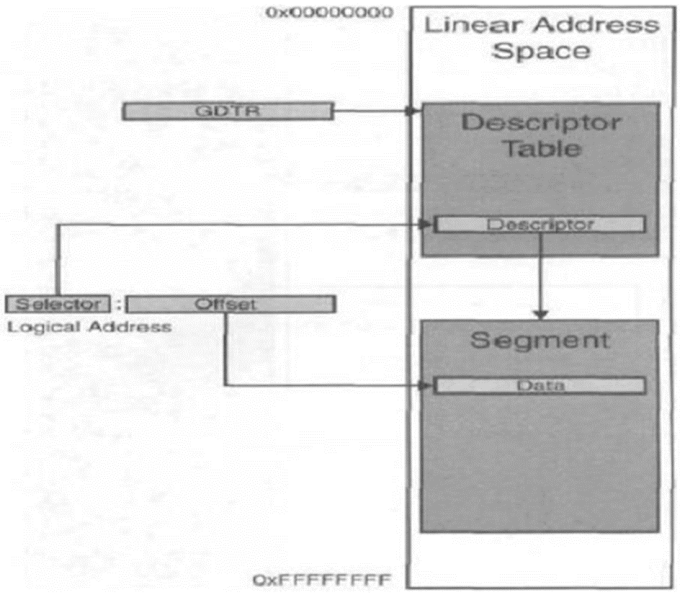

图3. 保护模式下寻址方式

上图就是保护模式下的寻址示意图。在保护模式下，当我们要访问内存中的数据、代码或一些其他的资源时，计算机并不是直接进行访问的，而是要经过一系列复杂的地址变换才能找到我们真正想访问的资源。

为什么要进行如此复杂的操作，原因就是保护内存中各个资源，解决内存访问的不安全性。例如在实模式下，我们可以随意访问并修改内存中1MB中的任何内容。但对于内存中一些重要的资源，如果就这样简单地修改了，就会造成不可逆转的危害。因此，在保护模式下的内存中，存在这样一块表，对内存中的资源分门别类，让相同的资源放在一块。这样，假设我们要访问内存中的数据，就需要查找该表，间接地访问内存中的数据，有效防止用户对其他资源进行修改，保护了内存的安全。这样的表在内存中一共有两种，分别是GDT（全局标识符表）和LDT（局部标识符表），全局描述符表只有一张，而局部描述符表有多张。这里我们以全局描述符表为例。上图右边的内存descriptor table就是全局描述符表。

在全局描述符表中，其每一项都是一种特殊的数据结构——段描述符，其在32位内存中占2个存储单元，如图一右边的descriptor就是段描述符。段描述符里面存放不同段的各种信息，如段的首地址，段的长度，段的访问权限，段的数据类型等等，如下图所示：

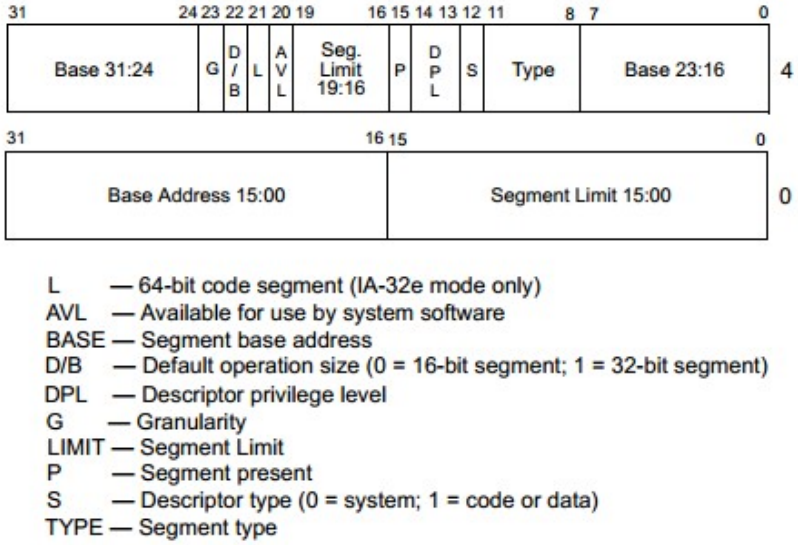

图4. 段描述符

我们根据段描述符中的信息就可以找到我们想要的段区间了，也就是图一右边的Segment。Eip中寄存器通常存放段的偏移量（offset），我们只需要将段描述符中段的基地址加上偏移量就可以访问我们想要访问的内存单元（data）了。

那么我们如何找到这样一张表的位置呢？GDTR提供了我们全局描述符表在内存中的首地址，而LDTR提供了我们局部描述符表在内存中的首地址。利用这两种寄存器，我们就可以轻松地找到表的位置了。

想到访问段的数据就需要先找到相应的段描述符，在描述符表中，段描述符有许多种，如代码段描述符、数据段描述符、栈段描述符。如何找到相应的的段描述符呢？这就需要我们段选择子起作用了。

进入32位保护模式，段寄存器 CS 、DS、 ES、 FS、 GS、 SS ，它们还是16位的，但是不再存着所谓的段地址了，而是成为段选择器，里面存着段选择子如下图所示：

图5.段选择子

3~15是段描述符在描述符中的索引，也就是相对段描述发表首地址的偏移量 TI = 0时表示描述符在GDT中，TI = 1时，描述符在LDT中，RPL是请求特权级，表示给出当前选择子的那个程序的特权级，正是该程序要求访问这个内存段.通过不同的段选择子，我们就可以找到相应的选择符了。

上述就是保护模式下段模式的寻址方式，在页模式下，还要进行分页访问，这里就不进行阐述了。

3、实模式中三种地址的关系

为了充分利用地址空间，采用：段基址+段偏移 的方式，对20位的地址空间进行寻址。(下一指令)物理地址=16CS+IP，其中CS存放段基址，IP存放段偏移，便恰好可以对20位地址空间进行寻址(对堆栈的访问则是SS:SP；对数据段的访问是DS:DI或DS:BX)。 以上就是分段机制，(段基址：段偏移)称为逻辑地址；(16段基址+段偏移)就是物理地址（分段机制中，物理地址就是线性地址）。

4、保护模式中三种地址的关系

在保护模式下，我们的偏移值从20位变成了32位，存放在eip寄存器下。其中段选择符(段基址)+偏移量就是逻辑地址；逻辑地址经过分段部件变换成为线性地址；如果不分页，得到的线性地址就是物理地址。如果分页，则线性地址要经过分页部件变换后才是物理地址。如图1所示，16位的段选择子：32位的偏移量就是逻辑地址，通过段选择符表找到内存中的data，这个data就是线性地址，该图并未采用分页结构，因此线性地址就是物理地址。

三、任务分工和完成情况

1. 胡峻豪：了解实模式，保护模式寻址方式及地址转换的基本情况，探讨研讨核心要点，PPT制作，实模式寻址及地址转换演讲，报告撰写。

2. 胡天磊：了解实模式，保护模式寻址方式及地址转换的基本情况，探讨研讨核心要点，保护模式寻址及地址转换演讲，报告撰写。

3. 胡才郁：了解实模式，保护模式寻址方式及地址转换的基本情况，探讨研讨核心要点，报告撰写。

4. 张俊雄：了解实模式，保护模式寻址方式及地址转换的基本情况，探讨研讨核心要点，报告撰写。

四、参考资料来源

1、https://blog.csdn.net/judyge/article/details/52336174

2、https://zhuanlan.zhihu.com/p/42309472

3、https://zhuanlan.zhihu.com/p/486369422

4、https://blog.csdn.net/weixin_42968829/article/details/100897095

五、附件信息

1、附件研讨PPT文档（pdf）。