# 2021111000 李卓凌 第一次读书笔记

# 1.1 系统级体系结构概览

## 1.描述符表

GDT：

全局描述符表，包括：段描述符、TSS描述符、LDT描述符

LDT：

局部描述符表，包括：段描述符、调用门

在保护模式下操作时，所有的内存访问都通过GDT或LDT；表里是段描述符，段描述符有与之对应的段选择符；要访问段中的内容，必须同时提供段选择符和偏移地址；GDT的线性地址存在GDTR中，LDT的线性地址存在LDTR中。

IDT：

中断描述符表，包括：中断门、任务门、陷进门

外部中断、软件中断、异常是通过IDT处理的

## 2.系统段

任务状态段（TSS）：

定义了任务执行环境的状态，包括通用寄存器、段寄存器、EFLAGES寄存器、EIP寄存器和段选择符以及三个堆栈段的指针的状态；

所有运行在保护模式下的程序，都在当前任务的上下文中进行；当前任务的TSS的段选择符保存在任务寄存器中；切换任务可以通过CALL或JMP(书中也给出了任务切换处理器的操作步骤)

LDT：

介绍同上

## 3.系统寄存器

控制寄存器：

CR0、CR2、CR4，包含了若干标志和数据域用于控制系统级的操作，这些寄存器内的其他标志指明了操作系统对处理器的兼容

任务寄存器：

包含了当前任务的TSS线性地址和界限

EFLAGES寄存器：

EFLAGES寄存器内的系统标志和IOPL域，控制着任务和模式切换、中断处理、指令跟踪和访问特权

调试寄存器、模式相关的寄存器等未列出

## 4.内存管理

直接物理地址：

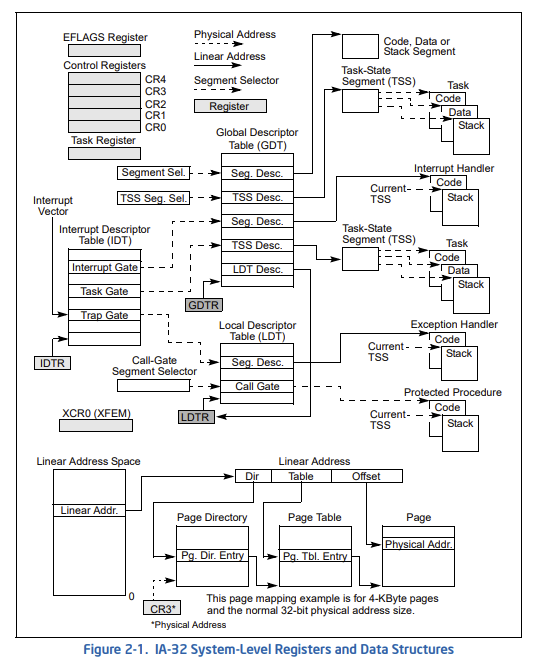
线性地址就是物理地址

虚拟内存：

使用分页机制，一个线性地址被分为页目录、页表、页框中的偏移量；当使用分页时，所有代码、堆栈、系统段、GDT、IDT都可以将最近访问过的页驻留在内存中，进行分页。页目录和页表是保存页的两个系统数据结构。

## 5.其他系统资源

操作系统指令、性能检测计数器、内部高速缓存和缓冲区



# 1.2 实模式和保护模式的转化

## 1.四种模式介绍

实模式：

提供了intel8086的编程模式和一些扩展

保护模式：

处理器的原生模式，涵盖了处理器所有的特点和指令，有着最好的性能；处理器上电或重启后自动进入实模式，控制寄存器CRO中的PE标志控制着处理器是在实模式还是保护模式下；PE=0，切换到实模式，PE=1，切换到保护模式。

系统管理模式：

为操作系统实现电源和OEM专有特征提供的一种透明的机制，接受到SMI则切换到SMM模式

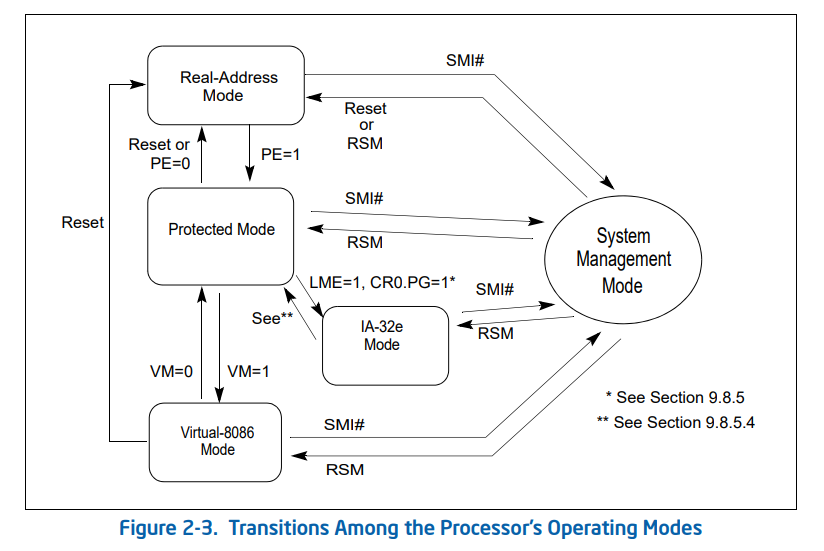
虚拟8086模式：

允许在多任务的保护模式下处理执行8086程序，VM标志决定处在保护模式还是虚拟8086模式

## 2.模式转换

这一节的模式转化讲的很粗，去看了9.9节mode switching：

保护模式通过执行 MOV CR0 这个指令切换，该指令将CR0寄存器中PE标志设为1；同理，将CR0中的PE标志执行clear操作实现切换回实模式。



# 1.3 80x86系统指令寄存器

## 1.EFLAGES寄存器：

ID 识别位：

表明是否支持CPUID指令

VIP虚拟中断等待：

置1表示有一个中断是正在等待被处理，置0表示没有

VIF虚拟中断标志:

虚拟中断包含了一个IF标志的虚拟映象。这个标志是和VIP标志一起使用的。当控制寄存器CR4中的VME或者PVI标志置为1且IOPL小于3时，处理器只识别VIF标志(VME标志用来启用虚拟8086模式扩展，PVI标志启用保护模式下的虚拟中断)。

VM虚拟8086模式:

置1进入虚拟8086模式，置0返回保护模式

RF恢复标志：

控制着处理器对断点指令条件的响应

NT嵌套任务标志：

控制被中断和被调用的任务的链接

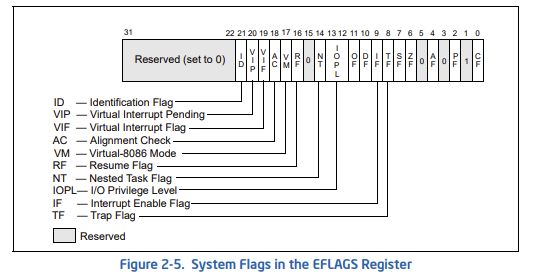
IOPL IO特权级：

指出当前程序或任务的I/O级别，当前程序或任务的CPL必须小于等于IOPL才可以访问I/O特权

IF中断启用标志：

控制着处理器对可屏蔽硬件中断请求的响应。置1是响应可屏蔽硬件中断，置0为禁止响应可屏蔽硬件中断，IF标志并不影响异常和不可屏蔽中断(NMI)的产生。控制寄存器CR4中的CPL、IOPL和VME标志决定着IF标志是可否可以由指令CLI、STTI、POPF、POPFD和IRET修改。

TF陷阱标志：



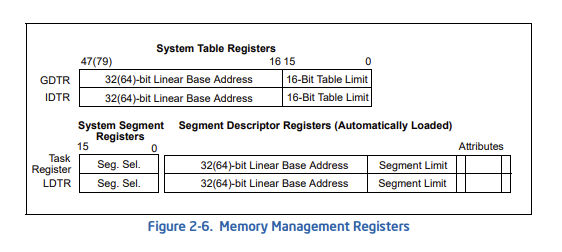
## 2.内存管理寄存器：GDTR、LDTR、IDTR、TR

GDTR: GDTR寄存器保存了GDT的32位基地址和16位表界限。基地址是指GDT的0字节的线性地址，表界限是指表中的字节个数。LGDT和SGDT指令是用来分别装载和保存GDTR寄存器的。处理器一上电或复位，基地址就被设为缺省的0，表界限设为FFFFH。对于保护模式的操作，作为处理器初使化过程的一部分，一个新的基地址必须装入GDTR。

LDTR: LDTR寄存器保存了16位段选择符、32位基地址、16位段界限和LDT描述符属性。基地址是指LDT段的0字节的线性地址，段界限是指段中的字节个数。LLDT和SLDT指令是专门分别用来装载和保存LDTR寄存器段选择符那部分的。包含LDT的段必须在GDT中有一个段描述符。当LLDT指令装载一个LDTR中的段选择符时，LDT描述符的基地址、界限和描述符属性就自动装载到LDTR中。当进行任务切换时，LDTR就会自动被装载连同新任务的段选择符和描述符。在写新的LDT信息到寄存器前，LDTR的内容前并不会自动的保存。处理器一上电或复位，段选择符和基地址都被设缺省的0

IDTR: IDTR寄存器保存了IDT的32位基地址和16位表界限。基地址是指IDT的字节0的线性地址，表界限是指表中的字节个数。LIDT和SIDT是专门分别用来装载和保存IDTR寄存器的指令。处理器一上电或复位，基地址就被设为缺省的0，界限就被设为FFFH。作为处理器初使化程的一部分，寄存器中的基地址和界限可以改变。

TR：任务寄存器保存着16位的段选择符，32位基地址，16位段界限和当前任务的TSS描述符属性。它引用GDT中的TSS描述符。基地址指明TSS中的0字节的线性地址，段界限指明TSS中的字节个数。



## 3.控制寄存器：CR0 CR1 CR2 CR3 CR4

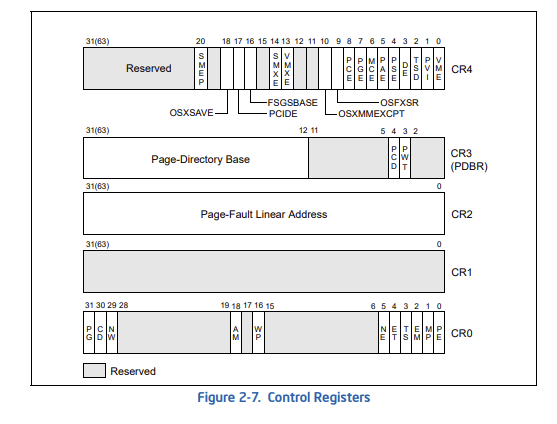
CR0:包含系统控制标志

CR1:保留

CR2：包含缺页的线性地址

CR3: 包含了页目录的基地址和第二个标志，PCD和PWT；该寄存器也被称为页目录基地址寄存器(PDBR)。页目录基地址只有高20位确定，低12位是0，所以页日录地址必须是页边界对齐的(4K字节)。PCD和PWT标志控制着页目录在处理器内部数据缓冲区的缓存（它们不控制TLB页目录信息的缓存)。当使用物理地址扩展时，CR3寄存器包含了页目录指针表的基地址

CR4：包含了一组标志，这些标志启用了架构方面的几个扩展，并指明了系统对某些处理器支持的能力。这个控制寄存器可以通过用MOV指令“从寄存器读或者写到寄存器”的方式进行读取或者装载（修改）。在保护模式下，MOV指令允许读取或者装载控制寄存器（在0级特权下)。这个限制意味着应用程序或者操作系统过程（运行在1、2、3级特权下）不能读取或者装载控制寄存器。装载控制寄存器时，保留位应该保持以前读取的值。



# 1.4系统指令

用来处理系统级的功能，比如装载系统寄存器、管理高速缓冲寄存器、管理中断或者设置调试寄存器。

LGDT: LGDT指令将GDT的入口地址装入GDTR寄存器，从此以后，CPU就根据此寄存器中的内容作为GDT的入口来访问GDT。GDTR中存放的是GDT在内存中的基地址和其表长界限。

SGDT: SGDT用于保存GDTR寄存器中的值。

LIDT、SIDT: 用于加载和保存LDTR寄存器的段描述符部分，即段基址和段限长度。

LLDT、SLDT: 用于加载和保存IDTR寄存器

LTR、STR: TR寄存器用于存放当前任务TSS段的16位段选择符、32位基地址、16位段长度和描述符属性值。指令LTR和STR分别用于加载和保存TR寄存器的段选择符部分。

下图是将书中的指令表翻译后的图：

