[Intel 80386 微处理器的系统结构](http://blog.csdn.net/scucj/article/details/4252906)

http://blog.csdn.net/scucj/article/details/4252906

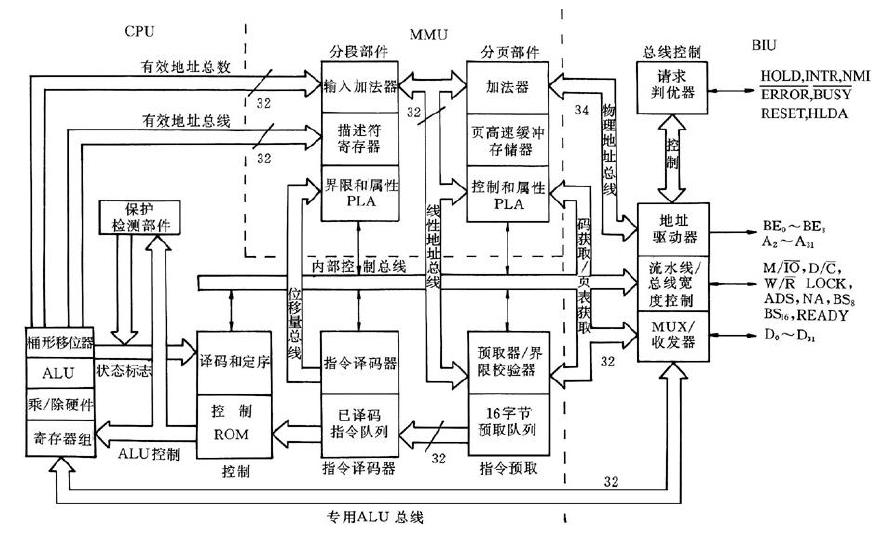
一、Intel 80X86微处理器系列  
      1.从8080/8085到8086  
      (1)8086 有16位寄存器和16位外部数据总线，20位地址总线，寻址为1MB的地址空间；8080/8085有只有8位，寻址地址空间为64K。  
      (2)8086有一个初级流水线结构，内部操作与对外操作具有并行性，即EU与BIU可分别独立工作，8085则没有这样的功能。  
      (3)具有里程碑的意义。

      2.从8086到8088   
      (1)8088 的外部数据总线为8位，8086片内指令预取缓冲器深度为6字节，8088只有4字节, 其他与8086是一样的。   
      (2)8088易与已有的8位外围芯片配合使用，且价格低，获得了巨大的成功。

      3.80286、80386、80486  
      (1)80286:16位结构，有16位寄存器和16位外部数据总线，24位地址总线,16MB寻址空间，支持保护方式，可执行多任务。  
      (2)80386:32位结构，内部寄存器与数据总线都是32位的，但是段寄存器仍为16位，32位地址总线，4GB寻址空间，支持保护虚地址方式，保留原有的实地址方式用于与8086兼容，可执行多任务。  
      (3)80486:32位结构，有一个与80386相同结构的主处理器, 另外集成了浮点处理部件(FPU)部件和一个8K字节的高速缓冲存储器(CACHE)，速度比80386快很多。

二、Intel 80386 微处理器的基本结构

Intel 80386 微处理器功能结构如图一，它是三部分组成：中央处理部件CPU、存储管理部件MMU、总线接口部件BIU。



图一 Intel 80386 微处理器功能结构

1.中央处理部件CPU：它由指令部件和执行部件组成。

(1)指令部件包括指令预取和指令译码两部分，各有一个队列，分别为指令预取队列和指令译码队列。指令预取用从存储器取出来的指令代码临时保存在指令预取队列中，然后见过指令译码的预译码后放入到已译码指令队列中，以供指令执行部件调用。执行部件由8个32位的通用寄存器、1个32位的算术运算单、1个64位的桶形移位器和乘/除硬件。80386采用微程序控制方式，所有80386指令对应的微程序存放在其内部的控制存储器中。指令译码器从指令预取队列中取指令，将操作码译成与该指令操作码相对应的控制ROM的入口地址，并存入译码指令队列中，该队列可容纳3条已译码指令。

(2)执行部件执行从译码指令队列中取出的已译码指令。它包含8个32位通用寄存器、32位的算术运算单元ALU、1个64位的桶形移位器和乘/除硬件。由于采用微程序技术，所以一条80386指令的执行过程实际是逐条执行该指令所对应的微程序中的微指令。出于缩短微指令宽度，即简化控制存储器电路的需要，微指令中广泛使用字段编译法，因此在行微指令的过程中还需要译码，即控制单元要对微指令的字段进行译码，产生一条指令操作的时序控制信号(即微操作信号)。80386的各个功能部件既能独立工作，又能与其他部件配合工作，因此80386可采用比8086并行度更高的流水线操作方式，如图二所示，从图中可以看出，总线接口部件、指令译码部件、执行部件及存储器管理部件四个部件并行工作，因此80386的指令流水线为4级。[1]



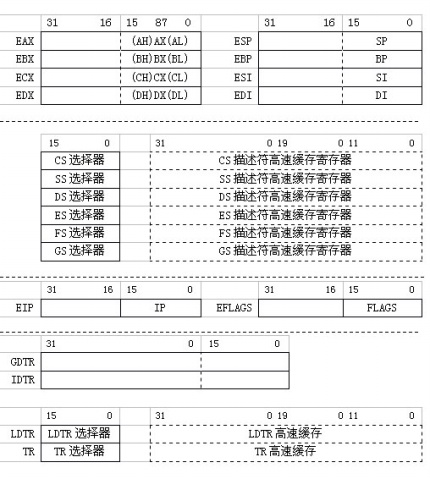
图二 Intel 80386 流水线操作示意图

2. 存储管理部件(MMU)由分段部件和分页机构组成，实现了从逻辑地址到物理地址的转换，既支持段式存储管理、页式存储管理，也支持段页存储管理。它存储器采用段、页式结构，80386首次将分页机制引入到80X86结构，每页大小为4KB。

3.总线接口部件功能：它通过数据总线、地址总线、控制总线来与外部环境联系，包括从存储器中预取指令、读写数据，从I/O端口读写数据，以及其他的控制功能。数据总线和地址总线都是32位的，由于它们是分开的，所以从存储器中存储数据最快也需要两个时钟周期内完成。

三、80386的寄存器结构

80386微处理器共有7类34个寄存器，通用寄存器组、段寄存器、指令指针和标志寄存器、系统地址寄存器、控制寄存器、调试寄存器、[**测试**](http://lib.csdn.net/base/softwaretest)寄存器。前四类寄存器的示意图三。

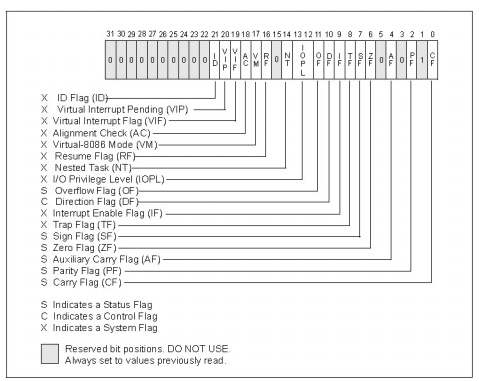


图三 通用寄存器组、段寄存器、指令指针和标志寄存器、系统地址寄存器示意图

(1)通用寄存器组：共有8个32位寄存器，EAX， EBX，ECX，EDX，ESP，EBP，ESI，EDI。它们由8086的16位寄存器扩展而来，它们的低16位与8086使用方法相同。

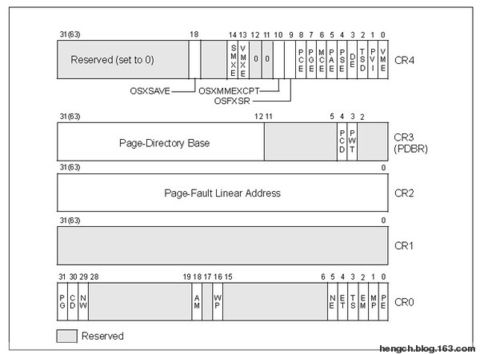
(2)段寄存器：共有6个16位的段寄存器CS、DS、SS、ES、FS、GS。与这6个段寄存器对应的有6个64位描述符寄存器，它是80X86处理器提供的一种附加的非编程的寄存器，用来装64的段描述符，每当一个段选择符被装入段寄存器是，相应的段描述符就由内存装入到对应的非编程的CPU寄存器。其中CS、DS、SS、ES与8086的段寄存器完全相同，在实地址方式下，使用方法也与8086相同；在虚地址保护方式下，这些寄存器中的值是“段选择符”，需要查全局描述符表（GDT）或者局部描述符表（LDT）来获得段的基地址，再加上偏移地址才能得到线性地址。 FS和GS是新加的附加数据段寄存器，可以由用户将FS、GS定义为其他数据段。

(3)指令指针和标志寄存器：指令指针寄存器EIP，由8086的IP寄存器扩展而来。标志寄存器EFLAGS包含一组状态标志、一个控制标志、一组系统标志，图四定义该寄存器中的标志位。



图四 标志寄存器EFLAGS的标准位

标志寄存器EFLAGS的低12位与8086的标志寄存器FLAGS一样。IOPL位表示特权标志位，定义当前任务的特权层。NT位表示任务嵌套标志位，当NT位为1时表明当前执行的任务嵌套在另外一个任务中，否则NT位为0。RF位表示重新启动标志位，与调试寄存器一起用于断点和单步操作，RF位为1时表明下一条指令的调试故障将被忽略，不产生中断异常；RF位位0时表示调试故障被接受并产生中断异常。由于调测失败后强迫程序恢复执行；在每条指令成功执行后，RF自动复位。VM位表示虚拟模式标志位，VM位为1时表明80386工作在保护虚拟地址方式。前4个定义从80286开始，后面的2个定义从80386开始存在。另外的三个标志是Pentium以后的CPU才有的。VIF(Virtual interrupt flag)表示虚拟中断标志。当VIF=1时，可以使用虚拟中断，当VIF=0时不能使用虚拟中断。该标志要和下面的VIP和CR4中的VME配合使用。VIP(Virtual interrupt pending flag)表示虚拟中断挂起标志。当VIP=1时，VIF有效，VIP=0时VIF无效。ID(Identification flag)表示鉴别标志。该标志用来只是Pentium CPU是否支持CPUID的指令。[2]  
(4) 系统地址寄存器和系统段寄存器：系统地址寄存器有全局描述符表寄存器GDTR、中断描述符表寄存器IDTR。系统段寄存器有局部描述符表寄存器LDTR和任务寄存器TR。这些寄存器保存相应的描述符表的地址。  
(5) 控制寄存器：4个32位的控制寄存器CR0，CR1，CR2，CR3，它们保存全局性的机器状态，其基本定义如图五。从Pentium开始，又增加了一个CR4。下面来简单介绍控制寄存器中的位。[2]



图五 控制寄存器的基本定义

1)CR0的低16位包含了与80286的MSW一致的位定义，保持了和80286的兼容，同时也兼容了从80286开始的两条指令LMSW/SMSW。指令LMSW和SMSW分别用于装入和保存机器状态字信息，可以通过MOV指令对CR0进行读写操作。CR0中各位含义如下：  
PE(Protection Enable)保护模式允许位，用来启动CPU进入虚地址保护方式。PE=0表示CPU工作在实地址方式；PE=1表示CPU工作在虚地址保护方式。  
MP(Monitor Coprocessor)监控协处理器，MP=1表示协处理器在工作；MP=0表示协处理器未工作。   
EM(Emulation)协处理器仿真，当MP=0，EM=1时，表示正在使用软件仿真协处理器工作。   
TS(Task Switched)任务转换，每当进行任务转换时，TS=1；任务转换完毕，TS=0。TS=1时不允许协处理器工作。    
ET(Extension Type)处理器扩展类型，反映了所扩展的协处理器的类型，ET=0为80287，ET=1为80387。   
PG(Paging)页式管理机制使能，PG=1时页式管理机制工作，否则不工作。  
NE(Numeric Error)数值异常中断控制，NE=1时，如果运行协处理器指令发生故障，则用异常中断处理，NE=0时，则用外部中断处理。   
WP(Write Protect)写保护，当WP=1时，对只读页面进行写操作会产生页故障。  
AM(Alignment Mask)对齐标志，AM=1时，允许对齐检查，AM=0时不允许，关于对齐，在EFLAGS的AC标志时介绍过，在80486以后的CPU中，CPU进行对齐检查需要满足三个条件，AC=1、AM=1并且当前特权级为3。   
NW(Not Write-through)和CD(Cache Disable)，这两个标志都是用来控制CPU内部的CACHE的，当NW=0且CD=0时，CACHE使能，其它的组合比较复杂。  
前4个定义从80286开始，接着的2个定义从80386开始存在, 后面4个是从80486开始定义的。

2)CR1寄存器用来保留给Intel微处理器将来开发使用；CR2寄存器包含一个32位的线性地址，指向发生最后一次也故障的地址，只有在PG=1时，CR2才有效，当页故障处理程序被激活时，压入页故障处理程序堆栈中的错误码提供页故障的状态信息;CR3寄存器中包含页物理目录表的物理基地址，由于每4KB为一页，80386中的页目录表总在页的整数边界上，CR3的低13位总是为0，只有当CR0中的PG=1时，CR3的页目录基地址才有效。  
(6) 调试寄存器：共8个排错寄存器DR0~DR7。DR0~DR3可以分别设置4个断点的线性地址，DR4~DR5保留未用，DR6是断点状态寄存器，DR7是断点控制寄存器(包括断点类型、断点长度，断点开放/禁止)。[2]  
(7) 测试寄存器：2个32位的测试寄存器TR6和TR7，用于控制转换后援缓冲器中的RAM测试，其中TR6为命令测试寄存器，TR7为测试数据寄存器。[2]

四、80386/80486微处理器的存储器管理

80386三种工作方式：实地址方式，虚地址保护方式，虚拟8086方式。由于这三种方式的介绍又可以写一篇blog了，这里就不详述了。

五、80486微处理器的的特点简介

(1)将80386处理器的指令译码和执行部件扩展成五级流水线。  
      (2)采用突发总线（Burst Bus）同内存进行高速数据交换。  
      (3)采用8KB统一的代码和数据CACHE(高速缓冲存储器)。  
      (4)首次将浮点处理部件80X87 FPU集成到了微处理器内。  
      (5)具有完整的RISC内核，指令执行单元采用RISC技术和流水线技术，大部分基本指令执行时间为1个时钟周期。  
      (6)内部的自测试功能包括执行代码和访问数据时的断点陷阱，广泛地测试片上逻辑、CACHE、分页转换CACHE。  
      (7)软件向下兼容。

参考及引用资料：

[1]<http://www.timesk.com/?action-viewnews-itemid-80945>

[2]http://hengch.blog.163.com/blog/static/107800672009029104144323/