尽管有很多关于Windows使用方面的书籍，但很少有书讲述它是如何工作的。不过，查阅《Microsoft Windows Internals，4th ed》（Russionvich 和 Solomon，2004）是其中最好的选择之一。该书描述的虽然是Windows XP，但大部分的描述还是准确的。就内部机制而言，Windows XP和Windows Vista是非常相近的。

而且，微软通过Windows学术计划为大学教员和学生提供对其有帮助的Windows内核信息。该计划会发布大部分Windows Server 2003内核源代码、Cutler团队的原始NT设计文档和一大套源自Windows Internals 书籍的表述资料。另外，Windows驱动工具也会提供大量内核工作信息，因为设备驱动器不仅使用I/O设备，还需要使用进程、线程、虚拟内存和进程间的通信等。

11.3.1 操作系统结构

Windows Vista 操作系统包括很多层，如图11-6所示。在以下章节我们将研究操作系统中工作于内核态的最底级层次。其中心就是NOTS内核层自身，当Windows启动时由ntoskrnl.exe加载。NTOS包括两层，executive（执行体）提供大部分的服务，另一个较小的层称为内核（kernel），负责实现基础线程计划和同步抽象，同时也执行陷入句柄中断以及管理CPU的其他方面。

将NTOS分为内核和执行体体现了NT 的VAX/VMS根源。VMS操作系统也是由Cutler团队设计的，可分为4个由硬件实施的层次：用户、管理程序、执行体和内核，与VAX处理机结构提供的4种保护模式一致。Intel CPU也支持这4种保护环，但是一些早期的NT处理机对此不支持，因此内核和执行体表现了由软件实施的抽象，同时VMS在管理者模式下提供的功能，如假脱机打印，NT是作为用户态服务提供的。

NT的内核态层如图11-13所示。NTOS的内核层在执行体层之上，因为它实现了从用户态到内核态转换的陷入和中断机制。图11-13所示的最顶层是系统库ntdll.dll，它实际工作于用户态。系统库包括许多为编译器运行提供的支持功能以及低级库，类似于UNIX中的libc。Ntdll.dll也包括了特殊码输入指针以支持内核初始化线程、分发异常和用户态的异步过程调用（Asynchronous Procedure Calls，APC）等。因为系统库对内核运行是必需的，所以每个由NTOS创建的用户态进程都具有相同固定地址描绘的ntdll。当NTOS初始化系统时，会创建一个局部目标并且记录下内核使用的ntdll输入指针地址。

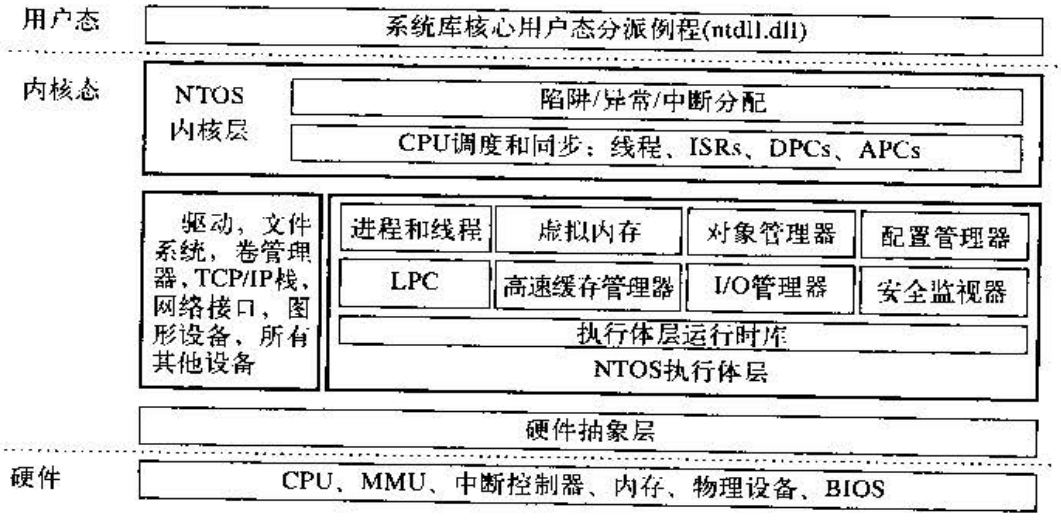

图11-13 Windows 内核态组织结构

在NTOS内核和执行体层之下是称为*硬件抽象层*（Hardware Abstraction Layer，HAL）的软件，该软件对类似于设备寄存器存取和DMA操作之类的底层硬件信息进行抽象，同时还就BIOS固件是如何表述配置信息和处理CPU芯片上的不同（如各种中断控制器）进行抽象。BIOS可以从很多公司获得，并且被集成为计算机母板上的永久内存。

内核态下另一个主要部件就是设备驱动器。Windows内核态下任何非NTOS或HAL的设备都会用到设备驱动器，包括文件系统、网络协议栈和其他如防病毒程序、DRM软件之类的内核扩展，以及与硬件总线接口的管理物理设备驱动器等。