# 中央处理器

ＣＰＵ即中央处理器。

中央处理器（CPU，Central Processing Unit）是一块超大规模的[集成电路](http://baike.baidu.com/view/1355.htm)，是一台[计算机](http://baike.baidu.com/view/3314.htm)的运算核心和控制核心。主要包括[运算器](http://baike.baidu.com/view/147768.htm)（ALU，Arithmetic and Logic Unit）和[控制器](http://baike.baidu.com/view/122229.htm)（[CU](http://baike.baidu.com/view/9259.htm)，[Control Unit](http://baike.baidu.com/view/2028057.htm)）两大部件。此外，还包括若干个[寄存器](http://baike.baidu.com/view/6159.htm)和[高速缓冲存储器](http://baike.baidu.com/view/496990.htm)及实现它们之间联系的[数据](http://baike.baidu.com/subview/38752/8058696.htm)、控制及状态的[总线](http://baike.baidu.com/view/1389.htm)。它与[内部存储器](http://baike.baidu.com/view/2397718.htm)和[输入](http://baike.baidu.com/view/700406.htm)/[输出](http://baike.baidu.com/view/700402.htm)设备合称为[电子计算机](http://baike.baidu.com/view/6373.htm)三大核心部件。

## 物理结构

CPU包括运算逻辑部件、[寄存器](http://baike.baidu.com/view/6159.htm)部件和控制部件等。

### 逻辑部件

英文Logic components；运算逻辑部件，可以执行[定点](http://baike.baidu.com/view/686843.htm)或[浮点](http://baike.baidu.com/view/924477.htm)算术运算操作、移位操作以及逻辑操作，也可执行[地址](http://baike.baidu.com/view/494802.htm)运算和转换。

### 寄存器

[寄存器](http://baike.baidu.com/view/6159.htm)部件，包括[通用寄存器](http://baike.baidu.com/view/1418486.htm)、专用寄存器和[控制寄存器](http://baike.baidu.com/view/4092816.htm)。

通用[寄存器](http://baike.baidu.com/view/6159.htm)又可分[定点数](http://baike.baidu.com/view/686808.htm)和[浮点数](http://baike.baidu.com/view/339796.htm)两类，它们用来保存指令执行过程中临时存放的[寄存器操作数](http://baike.baidu.com/view/1635040.htm)和中间（或最终）的操作结果。

通用[寄存器](http://baike.baidu.com/view/6159.htm)是中央[处理器](http://baike.baidu.com/view/50152.htm)的重要组成部分，大多数指令都要访问到通用寄存器。通用寄存器的[宽度](http://baike.baidu.com/view/51849.htm)决定[计算机](http://baike.baidu.com/view/3314.htm)内部的[数据通路](http://baike.baidu.com/view/1866929.htm)宽度，其[端口数目](http://baike.baidu.com/view/259082.htm)往往可影响内部操作的[并行性](http://baike.baidu.com/view/149970.htm)。

专用[寄存器](http://baike.baidu.com/view/6159.htm)是为了执行一些特殊操作所需用的寄存器。

控制[寄存器](http://baike.baidu.com/view/6159.htm)（CR0～CR3）用于控制和确定处理器的操作模式以及当前执行任务的特性。CR0中含有控制处理器操作模式和状态的系统控制标志；CR1保留不用；CR2含有导致页错误的[线性地址](http://baike.baidu.com/view/2690747.htm)；CR3中含有页目录表[物理内存](http://baike.baidu.com/view/138684.htm)基地址，因此该寄存器也被称为页目录基[地址寄存器](http://baike.baidu.com/view/178150.htm)PDBR（Page-Directory Base address Register）。

### 控制部件

英文Control unit；控制部件，主要是负责对指令[译码](http://baike.baidu.com/view/189742.htm)，并且发出为完成每条指令所要执行的各个操作的控制[信号](http://baike.baidu.com/view/54338.htm)。

其结构有两种：一种是以微存储为核心的[微程序](http://baike.baidu.com/view/854128.htm)控制方式；一种是以逻辑硬布线结构为主的控制方式。

微存储中保持[微码](http://baike.baidu.com/view/4883022.htm)，每一个微码对应于一个最基本的[微操作](http://baike.baidu.com/view/4694343.htm)，又称[微指令](http://baike.baidu.com/view/873614.htm)；各条指令是由不同序列的微码组成，这种微码序列构成微程序。中央[处理器](http://baike.baidu.com/view/50152.htm)在对指令[译码](http://baike.baidu.com/view/189742.htm)以后，即发出一定时序的[控制信号](http://baike.baidu.com/view/8407048.htm)，按给定序列的顺序以微[周期](http://baike.baidu.com/view/120466.htm)为节拍执行由这些微码确定的若干个[微操作](http://baike.baidu.com/view/4694343.htm)，即可完成某条指令的执行。

简单指令是由（3～5）个[微操作](http://baike.baidu.com/view/4694343.htm)组成，复杂指令则要由几十个微操作甚至几百个微操作组成。

## 主要功能

### 处理指令

英文Processing instructions；这是指控制程序中指令的执行顺序。程序中的各指令之间是有严格顺序的，必须严格按程序规定的顺序执行，才能保证[计算机](http://baike.baidu.com/view/3314.htm)系统工作的正确性。

### 执行操作

英文Perform an action；一条指令的功能往往是由[计算机](http://baike.baidu.com/view/3314.htm)中的部件执行一序列的操作来实现的。CPU要根据指令的功能，产生相应的操作[控制信号](http://baike.baidu.com/view/8407048.htm)，发给相应的部件，从而控制这些部件按[指令](http://baike.baidu.com/view/178461.htm)的要求进行动作。

### 控制时间

英文Control time；时间控制就是对各种操作实施时间上的定时。在一条[指令](http://baike.baidu.com/view/178461.htm)的执行过程中，在什么时间做什么操作均应受到严格的控制。只有这样，[计算机](http://baike.baidu.com/view/3314.htm)才能有条不紊地工作。

### 处理数据

即对数据进行算术运算和[逻辑运算](http://baike.baidu.com/view/397415.htm)，或进行其他的[信息处理](http://baike.baidu.com/view/553565.htm)。

其功能主要是解释[计算机指令](http://baike.baidu.com/view/786118.htm)以及处理[计算机软件](http://baike.baidu.com/view/8318.htm)中的[数据](http://baike.baidu.com/view/38752.htm)， 并执行指令。在微型计算机中又称微处理器，计算机的所有操作都受CPU控制，CPU的性能指标直接决定了[微机系统](http://baike.baidu.com/view/2955084.htm)的性能指标。CPU具有以下4个方面的基本功能：[数据通信](http://baike.baidu.com/view/49233.htm)，[资源共享](http://baike.baidu.com/view/8881.htm)，[分布式](http://baike.baidu.com/view/402382.htm)处理，提供系统[可靠性](http://baike.baidu.com/view/27210.htm)。运作原理可基本分为四个阶段：[提取](http://baike.baidu.com/view/2290232.htm)（Fetch）、[解码](http://baike.baidu.com/view/657399.htm)（Decode）、[执行](http://baike.baidu.com/view/278995.htm)（Execute）和写回（Writeback）。

## 工作过程

CPU从[存储器](http://baike.baidu.com/view/87697.htm)或[高速缓冲存储器](http://baike.baidu.com/view/496990.htm)中取出指令，放入[指令寄存器](http://baike.baidu.com/view/178142.htm)，并对指令[译码](http://baike.baidu.com/view/189742.htm)。它把指令分解成一系列的[微操作](http://baike.baidu.com/view/4694343.htm)，然后发出各种控制命令，执行微操作系列，从而完成一条指令的执行。指令是计算机规定执行操作的类型和[操作数](http://baike.baidu.com/view/420846.htm)的基本命令。指令是由一个[字节](http://baike.baidu.com/view/60408.htm)或者多个字节组成，其中包括[操作码](http://baike.baidu.com/view/178197.htm)字段、一个或多个有关[操作数](http://baike.baidu.com/view/420846.htm)[地址](http://baike.baidu.com/view/494802.htm)的字段以及一些表征[机器](http://baike.baidu.com/view/126147.htm)状态的状态字以及[特征码](http://baike.baidu.com/view/1184428.htm)。有的指令中也直接包含操作数本身。

### 提取

第一阶段，提取，从[存储器](http://baike.baidu.com/view/87697.htm)或[高速缓冲存储器](http://baike.baidu.com/view/496990.htm)中检索指令（为[数值](http://baike.baidu.com/view/3627588.htm)或一系列数值）。由程序[计数器](http://baike.baidu.com/view/259701.htm)（Program Counter）指定[存储器](http://baike.baidu.com/view/87697.htm)的位置。([程序计数器](http://baike.baidu.com/view/178145.htm)保存供识别程序位置的[数值](http://baike.baidu.com/view/3627588.htm)。换言之，[程序计数器](http://baike.baidu.com/view/178145.htm)记录了CPU在程序里的踪迹。)

### 解码

CPU根据[存储器](http://baike.baidu.com/view/87697.htm)提取到的指令来决定其[执行](http://baike.baidu.com/view/278995.htm)行为。在[解码](http://baike.baidu.com/view/657399.htm)阶段，指令被拆解为有意义的片断。根据CPU的[指令集架构](http://baike.baidu.com/view/1119651.htm)（ISA）定义将[数值](http://baike.baidu.com/view/3627588.htm)解译为指令。一部分的指令数值为运算码（Opcode），其指示要进行哪些运算。其它的[数值](http://baike.baidu.com/view/3627588.htm)通常供给指令必要的信息，诸如一个[加法](http://baike.baidu.com/view/56133.htm)（Addition）[运算](http://baike.baidu.com/view/1049334.htm)的[运算](http://baike.baidu.com/view/1049334.htm)目标。

### 执行

在提取和[解码](http://baike.baidu.com/view/657399.htm)阶段之后，紧接着进入执行阶段。该阶段中，连接到各种能够进行所需运算的CPU部件。

例如，要求一个[加法](http://baike.baidu.com/view/56133.htm)运算，[算术逻辑单元](http://baike.baidu.com/view/492754.htm)（ALU，Arithmetic Logic Unit）将会连接到一组输入和一组输出。输入提供了要相加的[数值](http://baike.baidu.com/view/3627588.htm)，而输出将含有总和的结果。ALU内含[电路](http://baike.baidu.com/view/134362.htm)系统，易于输出端完成简单的普通运算和[逻辑运算](http://baike.baidu.com/view/397415.htm)（比如[加法](http://baike.baidu.com/view/56133.htm)和位元运算）。如果[加法](http://baike.baidu.com/view/56133.htm)运算产生一个对该CPU处理而言过大的结果，在标志[暂存器](http://baike.baidu.com/view/238434.htm)里可能会设置运算[溢出](http://baike.baidu.com/view/98452.htm)（Arithmetic Overflow）标志。

### 写回

最终阶段，写回，以一定[格式](http://baike.baidu.com/view/33401.htm)将执行阶段的结果简单的写回。运算结果经常被写进CPU内部的[暂存器](http://baike.baidu.com/view/238434.htm)，以供随后指令快速存取。在其它案例中，运算结果可能写进速度较慢，但[容量](http://baike.baidu.com/view/334600.htm)较大且较便宜的主[记忆体](http://baike.baidu.com/view/167750.htm)中。某些类型的指令会操作[程序计数器](http://baike.baidu.com/view/178145.htm)，而不直接产生结果。这些一般称作“跳转”（Jumps），并在程式中带来循环行为、条件性执行（透过条件跳转）和函式。许多指令会改变标志[暂存器](http://baike.baidu.com/view/238434.htm)的状态位元。这些标志可用来影响程式行为，缘由于它们时常显出各种运算结果。例如，以一个“比较”指令判断两个值大小，根据比较结果在标志[暂存器](http://baike.baidu.com/view/238434.htm)上设置一个[数值](http://baike.baidu.com/view/3627588.htm)。这个标志可藉由随后跳转指令来决定程式动向。在执行指令并写回结果之后，[程序计数器](http://baike.baidu.com/view/178145.htm)值会递增，反覆整个过程，下一个[指令周期](http://baike.baidu.com/view/178156.htm)正常的提取下一个顺序指令。

## 性能参数

计算机的性能在很大程度上由[CPU](http://baike.baidu.com/view/2089.htm)的性能所决定，而CPU的性能主要体现在其运行程序的速度上。影响运行速度的性能指标包括CPU的工作频率、Cache容量、指令系统和逻辑结构等参数。

### 主频

主频也叫[时钟频率](http://baike.baidu.com/view/1087.htm)，单位是兆赫（[MHz](http://baike.baidu.com/view/809586.htm)）或千兆赫（[GHz](http://baike.baidu.com/view/1088055.htm)），用来表示CPU的运算、处理数据的速度。通常，主频越高，CPU处理数据的速度就越快。

CPU的主频=[外频](http://baike.baidu.com/view/1377.htm)×[倍频系数](http://baike.baidu.com/view/820849.htm)。主频和实际的运算速度存在一定的关系，但并不是一个简单的线性关系。　所以，CPU的主频与CPU实际的运算能力是没有直接关系的，主频表示在CPU内数字[脉冲信号](http://baike.baidu.com/view/717471.htm)震荡的速度。在Intel的[处理器](http://baike.baidu.com/view/50152.htm)产品中，也可以看到这样的例子：1 GHz Itanium芯片能够表现得差不多跟2.66 GHz[至强](http://baike.baidu.com/view/102947.htm)（[Xeon](http://baike.baidu.com/view/416484.htm)）/Opteron一样快，或是1.5 GHz Itanium 2大约跟4 GHz Xeon/Opteron一样快。CPU的运算速度还要看CPU的流水线、总线等各方面的性能指标。

### 外频

[外频](http://baike.baidu.com/view/1377.htm)是CPU的基准频率，单位是MHz。CPU的外频决定着整块[主板](http://baike.baidu.com/view/1143.htm)的运行速度。通俗地说，在[台式机](http://baike.baidu.com/view/1437416.htm)中，所说的[超频](http://baike.baidu.com/view/870.htm)，都是超CPU的外频（当然一般情况下，CPU的倍频都是被锁住的）相信这点是很好理解的。但对于[服务器CPU](http://baike.baidu.com/view/31389.htm)来讲，超频是绝对不允许的。前面说到CPU决定着主板的运行速度，两者是同步运行的，如果把服务器CPU超频了，改变了[外频](http://baike.baidu.com/view/1377.htm)，会产生异步运行，（台式机很多主板都支持异步运行）这样会造成整个[服务器系统](http://baike.baidu.com/view/6396042.htm)的不稳定。

绝大部分[电脑系统](http://baike.baidu.com/view/1356068.htm)中[外频](http://baike.baidu.com/view/1377.htm)与主板[前端总线](http://baike.baidu.com/view/1083.htm)不是同步速度的，而外频与前端总线（FSB）频率又很容易被混为一谈。

### 总线频率

前端总线（FSB)是将CPU连接到北桥芯片的总线。前端总线（FSB）频率（即[总线频率](http://baike.baidu.com/view/30203.htm)）是直接影响CPU与内存直接数据交换速度。有一条公式可以计算，即数据[带宽](http://baike.baidu.com/view/10821.htm)=（总线频率×数据[位宽](http://baike.baidu.com/view/1128.htm)）/8，数据传输最大带宽取决于所有同时传输的数据的宽度和传输频率。比方，支持64位的至强Nocona，前端总线是800MHz，按照公式，它的数据传输最大带宽是6.4GB/秒。

外频与前端总线（FSB）频率的区别：前端总线的速度指的是数据传输的速度，外频是CPU与主板之间同步运行的速 度。也就是说，100MHz外频特指数字脉冲信号在每秒钟震荡一亿次；而100MHz前端总线指的是每秒钟CPU可接受的数据传输量是 100MHz×64bit÷8bit/Byte=800MB/s。

### 倍频系数

倍频系数是指[CPU主频](http://baike.baidu.com/view/194487.htm)与外频之间的相对比例关系。在相同的外频下，倍频越高CPU的频率也越高。但实际上，在相同外频的前提下，高倍频的CPU本身意义并不大。这是因为CPU与系统之间[数据传输速度](http://baike.baidu.com/view/2827566.htm)是有限的，一味追求高主频而得到高倍频的CPU就会出现明显的“[瓶颈](http://baike.baidu.com/view/83216.htm)”效应－CPU从系统中得到数据的极限速度不能够满足CPU运算的速度。一般除了工程样版的Intel的CPU都是锁了倍频的，少量的如Intel[酷睿2](http://baike.baidu.com/view/304287.htm)核心的[奔腾双核](http://baike.baidu.com/view/1788253.htm)E6500K和一些至尊版的CPU不锁倍频，而[AMD](http://baike.baidu.com/view/810.htm)之前都没有锁，AMD推出了黑盒版CPU（即不锁倍频版本，用户可以自由调节倍频，调节倍频的超频方式比调节外频稳定得多）。

### 缓存

[缓存大小](http://baike.baidu.com/view/3749508.htm)也是CPU的重要指标之一，而且缓存的结构和大小对[CPU](http://baike.baidu.com/view/2089.htm)速度的影响非常大，CPU内缓存的运行频率极高，一般是和[处理器](http://baike.baidu.com/view/50152.htm)同频运作，[工作效率](http://baike.baidu.com/view/3456760.htm)远远大于[系统内存](http://baike.baidu.com/view/404470.htm)和硬盘。实际工作时，CPU往往需要重复读取同样的[数据块](http://baike.baidu.com/view/702806.htm)，而缓存[容量](http://baike.baidu.com/view/334600.htm)的增大，可以大幅度提升CPU内部读取数据的[命中率](http://baike.baidu.com/view/1792463.htm)，而不用再到内存或者硬盘上寻找，以此提高系统性能。但是由于CPU芯片面积和成本的因素来考虑，缓存都很小。

**L1　Cache**([一级缓存](http://baike.baidu.com/view/16882.htm)）是CPU第一层高速缓存，分为[数据缓存](http://baike.baidu.com/view/32571.htm)和[指令缓存](http://baike.baidu.com/view/1529137.htm)。内置的[L1高速缓存](http://baike.baidu.com/view/541082.htm)的[容量](http://baike.baidu.com/view/334600.htm)和结构对CPU的性能影响较大，不过[高速缓冲存储器](http://baike.baidu.com/view/496990.htm)均由静态RAM组成，结构较复杂，在CPU管芯面积不能太大的情况下，L1级高速缓存的容量不可能做得太大。一般服务器CPU的[L1缓存](http://baike.baidu.com/view/1211105.htm)的[容量](http://baike.baidu.com/view/334600.htm)通常在32－256KB。

**L2　Cache**（[二级缓存](http://baike.baidu.com/view/27650.htm)）是CPU的第二层高速缓存，分内部和外部两种芯片。内部的芯片二级缓存运行速度与主频相同，而外部的二级缓存则只有主频的一半。L2高速缓存[容量](http://baike.baidu.com/view/334600.htm)也会影响CPU的性能，原则是越大越好，以前家庭用CPU容量最大的是512KB，[笔记本电脑](http://baike.baidu.com/view/7690.htm)中也可以达到2M，而服务器和[工作站](http://baike.baidu.com/view/7977.htm)上用CPU的L2高速缓存更高，可以达到8M以上。

**L3　Cache**([三级缓存](http://baike.baidu.com/view/84602.htm)），分为两种，早期的是外置，[内存延迟](http://baike.baidu.com/view/1060639.htm)，同时提升[大数据](http://baike.baidu.com/view/6954399.htm)量计算时[处理器](http://baike.baidu.com/view/50152.htm)的性能。降低内存延迟和提升[大数据](http://baike.baidu.com/view/6954399.htm)量计算能力对游戏都很有帮助。而在服务器领域增加L3缓存在性能方面仍然有显著的提升。比方具有较大L3缓存的配置利用[物理内存](http://baike.baidu.com/view/138684.htm)会更有效，故它比较慢的[磁盘](http://baike.baidu.com/view/157418.htm)I/O子系统可以处理更多的[数据](http://baike.baidu.com/view/38752.htm)请求。具有较大L3缓存的[处理器](http://baike.baidu.com/view/50152.htm)提供更有效的[文件系统](http://baike.baidu.com/view/266589.htm)缓存行为及较短消息和处理器队列长度。

其实最早的L3缓存被应用在AMD发布的K6-III[处理器](http://baike.baidu.com/view/50152.htm)上，当时的L3缓存受限于[制造工艺](http://baike.baidu.com/view/1146582.htm)，并没有被集成进芯片内部，而是集成在主板上。在只能够和[系统总线频率](http://baike.baidu.com/view/3189424.htm)同步的L3缓存同主内存其实差不了多少。后来使用L3缓存的是[英特尔](http://baike.baidu.com/view/15281.htm)为服务器市场所推出的Itanium[处理器](http://baike.baidu.com/view/50152.htm)。接着就是P4EE和至强MP。Intel还打算推出一款9MB L3缓存的Itanium2[处理器](http://baike.baidu.com/view/50152.htm)，和以后24MB L3缓存的[双核心](http://baike.baidu.com/view/13560.htm)Itanium2处理器。

但基本上L3缓存对[处理器](http://baike.baidu.com/view/50152.htm)的性能提高显得不是很重要，比方配备1MB L3缓存的Xeon MP处理器却仍然不是Opteron的对手，由此可见前端总线的增加，要比缓存增加带来更有效的性能提升。

## 处理技术

在解释超流水线与超标量前，先了解流水线（Pipeline）。流水线是Intel首次在486芯片中开始使用的。流水线的工作方式就象[工业生产](http://baike.baidu.com/view/1714883.htm)上的装配流水线。在CPU中由5－6个不同功能的[电路](http://baike.baidu.com/view/134362.htm)单元组成一条指令处理流水线，然后将一条X86指令分成5－6步后再由这些电路单元分别执行，这样就能实现在一个[CPU时钟周期](http://baike.baidu.com/view/1876127.htm)完成一条指令，因此提高CPU的运算速度。经典奔腾每条整数流水线都分为四级流水，即[指令预取](http://baike.baidu.com/view/3658731.htm)、[译码](http://baike.baidu.com/view/189742.htm)、执行、写回结果，[浮点](http://baike.baidu.com/view/924477.htm)流水又分为八级流水。[超标量](http://baike.baidu.com/view/1062226.htm)是通过内置多条流水线来同时执行多个[处理器](http://baike.baidu.com/view/50152.htm)，其实质是以空间换取时间。而[超流水线](http://baike.baidu.com/view/326793.htm)是通过细化流水、提高主频，使得在一个[机器周期](http://baike.baidu.com/view/713240.htm)内 完成一个甚至多个操作，其实质是以时间换取空间。例如Pentium 4的流水线就长达20级。将流水线设计的步（级）越长，其完成一条指令的速度越快，因此才能适应工作主频更高的CPU。但是流水线过长也带来了一定副作 用，很可能会出现主频较高的CPU实际运算速度较低的现象，Intel的[奔腾4](http://baike.baidu.com/view/502769.htm)就出现了这种情况，虽然它的主频可以高达1.4G以上，但其运算性能却远远比不上AMD 1.2G的[速龙](http://baike.baidu.com/view/261293.htm)甚至奔腾III。

[CPU封装](http://baike.baidu.com/view/52049.htm)是采用特定的材料将CPU芯片或CPU[模块](http://baike.baidu.com/view/22845.htm)固化在其中以防损坏的保护措施，一般必须在封装后CPU才能交付用户使用。CPU的[封装方式](http://baike.baidu.com/view/3473353.htm)取决于CPU安装形式和器件集成设计，从大的分类来看通常采用Socket[插座](http://baike.baidu.com/view/294606.htm)进行安装的CPU使用PGA（栅格阵列）方式封装，而采用Slot x槽安装的CPU则全部采用SEC（单边接插盒）的形式封装。还有PLGA(Plastic Land Grid Array）、OLGA(Organic Land Grid Array）等[封装技术](http://baike.baidu.com/view/34754.htm)。由于市场竞争日益激烈，[CPU封装技术](http://baike.baidu.com/view/816474.htm)的发展方向以节约成本为主。

### 多线程

同时多[线程](http://baike.baidu.com/view/1053.htm)Simultaneous Multithreading，简称[SMT](http://baike.baidu.com/view/10599.htm)。SMT可通过复制[处理器](http://baike.baidu.com/view/50152.htm)上的结构状态，让同一个处理器上的多个[线程同步](http://baike.baidu.com/view/6501899.htm)执行并共享处理器的执行资源，可最大限度地实现宽发射、乱序的[超标量](http://baike.baidu.com/view/1062226.htm)处理，提高处理器运算部件的利用率，缓和由于数据相关或Cache未命中带来的访问内存延时。当没有多个线程可用时，SMT[处理器](http://baike.baidu.com/view/50152.htm)几乎和传统的宽发射[超标量](http://baike.baidu.com/view/1062226.htm)处理器一样。SMT最具吸引力的是只需小规模改变[处理器核心](http://baike.baidu.com/view/3957.htm)的设计，几乎不用增加额外的成本就可以显著地提升效能。[多线程技术](http://baike.baidu.com/view/2476093.htm)则可以为高速的运算核心准备更多的待处理数据，减少运算核心的闲置时间。这对于[桌面](http://baike.baidu.com/view/79807.htm)低端系统来说无疑十分具有吸引力。Intel从3.06GHz Pentium 4开始，部分[处理器](http://baike.baidu.com/view/50152.htm)将支持SMT技术。

### 多核心

[多核心](http://baike.baidu.com/view/757340.htm)，也指[单芯片多处理器](http://baike.baidu.com/view/1062228.htm)（Chip Multiprocessors，简称CMP）。CMP是由[美国斯坦福大学](http://baike.baidu.com/view/282776.htm)提出的，其思想是将大规模并行处理器中的SMP（对称多处理器）集成到同一芯片内，各个处理器[并行执行](http://baike.baidu.com/view/160033.htm)不同的进程。这种依靠多个CPU同时并行地运行程序是实现超高速计算的一个重要方向，称为并行处理。与CMP比较，SMT[处理器](http://baike.baidu.com/view/50152.htm)结构的灵活性比较突出。但是，当[半导体](http://baike.baidu.com/view/19928.htm)工艺进入0.18微米以后，线延时已经超过了门延迟，要求[微处理器](http://baike.baidu.com/view/1125.htm)的设计通过划分许多规模更小、局部性更好的[基本单元](http://baike.baidu.com/view/693012.htm)结构来进行。相比之下，由于CMP结构已经被划分成多个[处理器](http://baike.baidu.com/view/50152.htm)核来设计，每个核都比较简单，有利于优化设计，因此更有发展前途。IBM 的Power 4芯片和Sun的MAJC5200芯片都采用了CMP结构。[多核处理器](http://baike.baidu.com/view/2797908.htm)可 以在处理器内部共享缓存，提高缓存利用率，同时简化多处理器系统设计的复杂度。但这并不是说明，核心越多，性能越高，比如说16核的CPU就没有8核的 CPU运算速度快，因为核心太多，而不能合理进行分配，所以导致运算速度减慢。在买电脑时请酌情选择。2005年下半年，Intel和AMD的新型[处理器](http://baike.baidu.com/view/50152.htm)也将融入CMP结构。新[安腾处理器](http://baike.baidu.com/view/388413.htm)开发代码为Montecito，采用[双核心](http://baike.baidu.com/view/13560.htm)设计，拥有最少18MB片内缓存，采取90nm工艺制造。它的每个单独的核心都拥有独立的L1，L2和L3 cache，包含大约10亿支[晶体管](http://baike.baidu.com/view/30363.htm)。