

# 第 3 章

## CPU系统结构与故障维修

CPU（中央处理单元）也称为微处理器（Microprocessor）或处理器（Processor），它是计算机的核心部件，CPU 性能的高低直接反映了计算机的基本性能。

### 3.1 CPU 类型与组成

#### 3.1.1 CPU 的发展

1971 年 11 月，Intel 公司的 Ted Hoff（特德·霍夫）设计了世界上第一个微处理器 Intel 4004（如图 3-1 所示），这是一个里程碑式的产品。Intel 4004 是一款 4 位微处理器，指令和数据分开存储，分别有 4KB 指令缓存单元和 1KB 数据缓存单元。Intel 4004 CPU 有 45 条指令，工作时钟为 740kHz，能执行 4 位运算，支持 8 位指令集及 12 位地址，每秒运算 5 万次。它采用 P 沟道 MOS 晶体管技术制造，核心大小为 3mm×4mm，芯片集成了 2250 个晶体管，制程线宽为 10μm，工作电压为 15V，市场售价为 200 美元左右。4004 CPU 原先是为一家名为 Busicom 的日本公司而设计的，用来生产 Busicom 电算机。

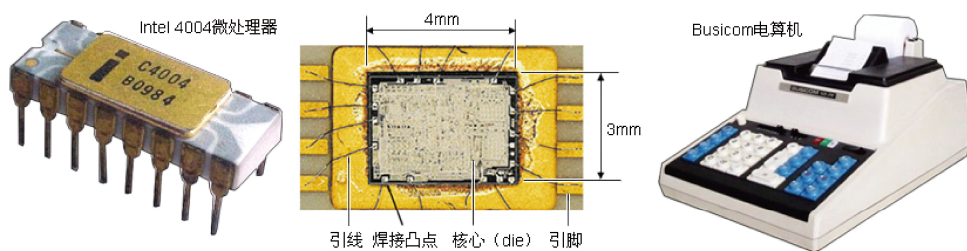


图 3-1 Intel 4004 CPU 和 Busicom 电算机（1971 年）

CPU 按产品市场可分为 x86 系列和非 x86 系列。目前 x86 系列 CPU 生产厂商只有 Intel、AMD、VIA 三家公司，x86 系列 CPU 在操作系统一级相互兼容，产品覆盖了 90% 以上的桌面计算机市场。非 x86 系列 CPU 生产厂商有 IBM、Sun、HP、ARM、MIPS、日立、三星、现代、中国科学院计算研究所等企业和单位。非 x86 系列 CPU 主要用于大型服务器和嵌入式系统，这些产品大多互不兼容，在桌面计算机市场中占有份额极小。Intel 公司 CPU 技术发展如表 3-1 所示。

表 3-1 Intel 公司主要 CPU 产品技术发展

CPU 系列	微结构代码	研发代号	工作频率	制程线宽	晶体管数	首次发布日期
4004	—	—	740kHz	10 $\mu$ m	2250	1971 年 11 月
8086	—	—	4.77MHz	3 $\mu$ m	2.9 万	1979 年 6 月
80286	—	—	6MHz	1.5 $\mu$ m	13.4 万	1982 年 2 月
80386	—	—	16MHz	1.5 $\mu$ m	27.5 万	1985 年 10 月
80486	—	—	33MHz	1.0 $\mu$ m	120 万	1989 年 6 月
Pentium	P5	—	60MHz	0.8 $\mu$ m	310 万	1993 年 3 月
Pentium Pro	P6	—	150MHz	0.50 $\mu$ m	550 万	1995 年 11 月
Pentium II	P6	Klamath	233MHz	0.35 $\mu$ m	750 万	1997 年 5 月
Pentium III	P6	Katmai	450MHz	0.25 $\mu$ m	950 万	2000 年 2 月
Pentium 4	Netburst	Willamette	1.5GHz	0.18 $\mu$ m	4200 万	2000 年 11 月
Pentium 4	Netburst	Northwood	2.0GHz	0.13 $\mu$ m	5500 万	2002 年 1 月
Pentium 4	Netburst	Prescott	2.8GHz	90nm	1.25 亿	2004 年 2 月
Pentium D	Netburst	Smithfield	2.8GHz	90nm	1.1 亿	2005 年 5 月
Pentium EE	Netburst	Smithfield	3.2GHz	90nm	2.30 亿	2005 年 11 月
Core Duo	Core	Yonah	2.16GHz	65nm	1.51 亿	2006 年 1 月
Core 2	Core	Merom	2.9GHz	65nm	2.91 亿	2006 年 7 月
Core i7	Nehalem	Bloomfield	3.2GHz	45nm	7.31 亿	2008 年 11 月
Core i5	Nehalem	Lynnfield	2.13GHz	45nm	3.82 亿	2009 年 8 月
Core i3	Nehalem	Clarkdale	3.33GHz	32nm	5.59 亿	2010 年

### 3.1.2 CPU 的类型

#### 1. Intel 公司 CPU 产品类型

Intel 公司的 CPU 产品按照市场应用可分为桌面型、移动型、服务器型和嵌入式 4 大类型；每个类型又分为不同的大系列和子系列。如桌面型 CPU 产品有 Core（酷睿）系列、Pentium（奔腾）系列、Celeron（赛扬）系列等，奔腾系列和赛扬系列产品属于早期产品，目前正在逐步淡出市场。酷睿系列 CPU 是 Intel 公司目前主推的产品系列，酷睿系列分为桌面型和移动型两类产品，并经历了 Core、Core 2、Core i5、Core i7 几代产品的发展。不同系列的 CPU 产品在外观上没有太大差别，但是设计技术和生产工艺各不相同。除嵌入式系列外，Intel 其他系列的 CPU 产品在软件上相互兼容。Intel 公司的 CPU 产品有十多个大系列，几百个子系列，上千个型号，典型产品如表 3-2 所示。

桌面型 CPU 是 Intel 公司的主推产品，很多新设计和新工艺都是在桌面型产品上取得

成功后,再推广到其他系列产品中。桌面型 CPU 产品主要用于商业办公和家庭个人计算机。

移动型 CPU 主要用于笔记本计算机和平板电脑。移动型 CPU 的性能要低于桌面型 CPU,移动型 CPU 的主要特点是低功耗和低发热。

服务器型 CPU 的性能高于桌面型产品,但是价格大大高于桌面型产品。服务器型 CPU 产品的特点是可以利用多个 CPU 组成多处理器系统。

嵌入式 CPU 主要用于工业控制计算机(如 Intel 8051)和智能手机(如 Intel PXA272)。

表 3-2 Intel 公司主要 CPU 产品系列

产品类型	产品系列	产 品 型 号			
		45nm	65nm	90nm	130nm
桌面型	酷睿系列	Core i7[4] Core2 Quad[4] Core2 Duo[2]	Core2 Quad[4] Core2 Duo[2]	Core [2] Core2 Duo[2]	N/A
	奔腾系列	Pentium E5200[2]	Pentium EE[2] Pentium D[2]	Pentium D[2] Pentium 4[1]	Pentium 4[1] Pentium III [1]
	赛扬系列	N/A	Celeron D[1]	Celeron[1]	Celeron[1]
移动型	移动酷睿	Core2 Quad M[4] Core2 Duo M[2] Core2 Solo[1/2]	Core2 Quad M[2] Core2 Duo M[2] Core Solo[1]	N/A	N/A
	凌动系列	Atom N230 系列[1] Atom N330 系列[2] Atom Z500 系列[1]	N/A	N/A	N/A
	迅驰系列	N/A	Centrino 2[1]	Centrino[1]	N/A
	移动奔腾	NA	N/A	Pentium M[1]	Pentium M[1]
	移动赛扬	Celeron M[1]	Celeron M[1]	Celeron M[1]	Celeron M[1]
服务器型	至强系列	Xeon [2/4/6/8]	Xeon[2]	Xeon[1]	Xeon[1]
	安腾系列	N/A	Itanium 2[2]	Itanium 2[2]	Itanium[1] Itanium 2[1/2]

注: [ ]内数值为 CPU 内核数; N/A 为没有相应产品。

## 2. AMD 公司 CPU 产品类型

Intel 与 AMD 公司的 CPU 虽然在性能和软件兼容性方面不相上下,但配套的硬件平台并不能相互完全兼容。例如,它们需要不同的主板进行产品配套。AMD 公司目前典型的 CPU 产品如表 3-3 所示。

AMD Phenom 系列,中文商标为“羿龙”。有 Phenom II、Phenom 两个子系列的产品,有 4 内核、3 内核的产品,是 AMD 主推产品。

AMD Athlon 系列,中文商标为“速龙”。有 Athlon X2、Athlon FX、Athlon 等子系列。

AMD Sempron 系列,中文商标为“闪龙”,是 AMD 系列低端产品。

AMD 笔记本系列 CPU 中文商标为“炫龙”,产品分类较为繁杂。

AMD Opteron 系列，中文商标为“皓龙”，主要用于服务器产品。

表 3-3 AMD 公司 CPU 产品系列

产品类型	产 品 型 号			
	45nm (K10)	65nm (K8L)	90nm (K8)	130nm (K7)
桌面型	Phenom II X4[4/6] Athlon II X2[2]	Athlon X2[2] Phenom X2[2]	Athlon 64 X2[2] Athlon 64[1]	Athlon 64 [1]
移动型	Turion II [2] Athlon II [2]	Turion X2[2] Athlon 64 X2[2]	Turion 64[1] Mobile[1]	Mobile[1]
服务器型	Opteron[4]	Opteron[2]	Opteron[2]	Opteron[1]

注：[ ]内数值为 CPU 内核数。

### 3.1.3 CPU 型号标识

#### 1. CPU 型号标识方法

如图 3-2 所示，Intel CPU 在金属外壳上蚀刻了几行字母，它们的具体含义如下。

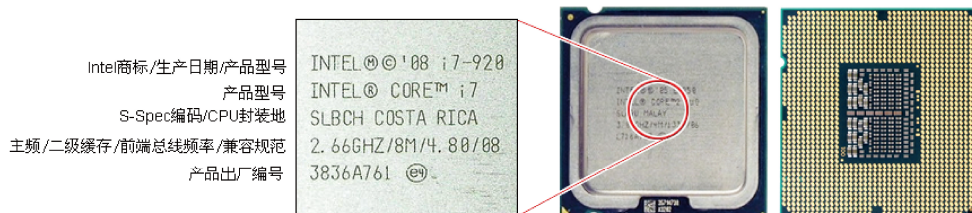


图 3-2 Intel CPU 产品标记

第 1 行按“Intel 商标/生产日期/产品型号”进行标记，如 INTEL 08 i7-920。

第 2 行标记为产品型号，如 INTEL CORE i7。

第 3 行为 S-Spec 编码和 CPU 封装地。CPU 产地有 COSTA RICA(哥斯达黎加), MALAY (马来西亚), CHINA (中国) 等。如标记为“SLBCH COSTA RICA”时，“SLBCH”是 S-Spec 编码，“COSTA RICA”表示 CPU 芯片在哥斯达黎加最后封装。

第 4 行为 CPU 技术参数，按“主频/二级或三级缓存/前端总线频率/兼容规范”顺序排列。如“2.66GHZ/8M/4.80/08”的标记表示：CPU 主频 2.66GHz，CPU 三级高速缓存 8MB，前端总线频率 4.8GT/s，08 为平台兼容标准。不同型号的 CPU，这一行参数各不相同。

第 5 行是产品出厂编号，Intel 公司没有公布编号的具体含义。

#### 2. S-Spec 编码方法

S-Spec 编码是 Intel 公司为方便用户查询 CPU 产品所制定的一组编码，它标注在 CPU 包装盒的标签和 CPU 金属外壳上（如图 3-2 所示）。S-Spec 编码很好识别，它们为 5 位，以“SL×××”进行标记，“×××”为英文字母或数字。没有 S-Spec 编码的 CPU 大多是工程测试样品。

通过 Intel 官方网站 (<http://processorfinder.intel.com/Default.aspx>), 可以直接查到 S-Spec 编码 CPU 的相关性能参数和技术文档 (如图 3-3 所示)。这些技术参数包括 CPU 主频、CPU 电源标准、前端总线频率、CPU 插座形式、制程工艺、发热功率、工作电压等。大部分技术参数很好理解, 个别参数的含义如下。

PCG (平台兼容性指南) 参数是 CPU 发挥正常功能所必需的电源标准版本, PCG 参数还提供哪种 CPU 可用于哪种主板的简便识别方法。这个标准仅适用于桌面 CPU, 笔记本 CPU 和服务器 CPU 均无此参数。



图 3-3 Intel Core i7 CPU 的 S-Spec 编码实例

### 3. 工艺步进

工艺步进编号表示 CPU 设计或制造的修订版本。工艺步进参数使用“字母+数字”表示。一般英文字母和数字越大, CPU 核心工艺就越新, 也意味着 CPU 工作更稳定, 例如, Intel Core2 E8500 CPU, 有“E0”和“C0”两种工艺步进, 而 E0 核心要比 C0 核心更稳定。

CPU 从设计到最终推向市场, 会经历无数苛刻的系统测试。即使如此, 有些细小的问题还是要到实际应用的时候才能发现。对于在使用中发现的问题, Intel 公司会推出新步进的 CPU, 解决已经发现的问题。其次, CPU 制造工艺一直处于不停的升级之中, 这些升级未必是很大的改进, 而是局部的, 某些细节工艺的提升, 这种情况也会使得 CPU 采用新的步进工艺。仔细比较就会发现, 工艺步进实际上与某款特定型号的 CPU 无关, 一款特定步进的晶圆可以应用在多款 CPU 核心制造上。

CPU 制造工艺的逐步提升和硬件纠错是工艺步进提升的原因。通常来说, 新步进的 CPU 超频能力更强, 发热也会略低。如果两颗 CPU 型号相同, 但工艺步进不同, 从 CPU 超频角度看, CPU 升级步进工艺的同时, 一般也会提高 CPU 的超频能力。提高 CPU 性能的方法如图 3-4 所示。

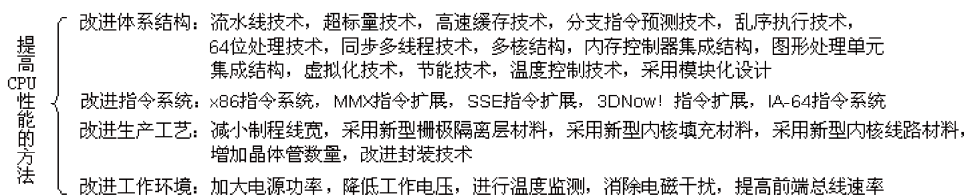


图 3-4 提高 CPU 性能的基本方法

### 3.1.4 CPU基本组成

#### 1. CPU基本组成

大部分CPU采用LGA或FC-PGA封装形式。FC-PGA封装是将CPU核心封装在基板上，这样可以缩短连线，并有利于散热。LGA采用无针脚触点封装形式。如图3-5所示，CPU由半导体硅芯片、基板、针脚或无针脚触点、导热材料、金属外壳等部件组成。

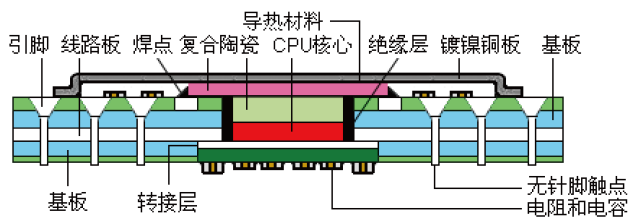


图 3-5 CPU 基本组成剖面图

(1) 外壳 (IHS)。CPU 金属外壳采用镀镍铜板，它的作用是保护 CPU 核心不受外力的损坏。外壳表面非常平整光滑，这有利于与 CPU 散热片的良好接触。

(2) 导热材料 (TIM)。在金属外壳内部与复合陶瓷之间，填充了一层导热材料，导热材料一般采用导热膏，它具有良好的绝缘性和极佳的导热性能，它的功能是将 CPU 内核发出的热量传导到金属外壳上。

(3) CPU 核心 (die)。CPU 核心是一个薄薄的硅晶片，尺寸一般为  $12\text{mm}\times 12\text{mm}\times 1\text{mm}$  左右。目前 CPU 核心中有多个内核 (2/4/6/8 个)，8 内核的 Intel Xeon CPU 集成的晶体管数达到了 24 亿个。

(4) 转接层。CPU 核心与基板之间有一个转接层，它的作用有三个：一是将非常细小的 CPU 内核信号线转接到 CPU 针脚上；二是保护脆弱的 CPU 核心不受损伤；三是将 CPU 核心固定在基板上。转接层采用复合材料制造，有良好的绝缘性能和导热性能。在转接层上，采用光刻电路与 CPU 内核的电路直接相连。在转接层下面，采用焊点与基板上的线路相连。

(5) 基板。金属封装壳周围是 CPU 基板，基板的功能一是连接转接层与 CPU 针脚，另外一个功能是设计一些电路，防止 CPU 内核的高频信号对主板产生干扰。

(6) 电阻和电容。基板底部中间有的电容和电阻，主要用于消除 CPU 对外部电路的干扰，以及与主板电路进行阻抗匹配。每个系列的 CPU 产品，这些电容和电阻的排列方式都有所不同。

(7) 针脚。基板下面的镀金无针脚触点，是 CPU 与外部电路连接的通道。

#### 2. CPU 内核组成

Core i7 系列 CPU 内核如图 3-6 所示，Core i7 CPU 一共有 4 个物理内核，内核面积为  $18.9\text{mm}\times 13\text{mm}=246\text{mm}^2$ 。每个 CPU 核心可以同时支持两个线程运行。每个核心都具有单独的 L1 与 L2 级高速缓存，同时 4 个核心使用共享的 L3 级高速缓存。在 Core i7 CPU 内部首次引入了 IMC (集成内存控制器) 和 QPI (快速路径互连) 总线。

Core i7 CPU 内核分为核心（Core）与非核心（UnCore）两大部分。核心部分包括 CPU 执行流水线和 L1、L2 级高速缓存。非核心部分为 L3 级高速缓存、集成内存控制器（IMC）、快速路径互连总线（QPI），以及功耗与时钟控制单元等。

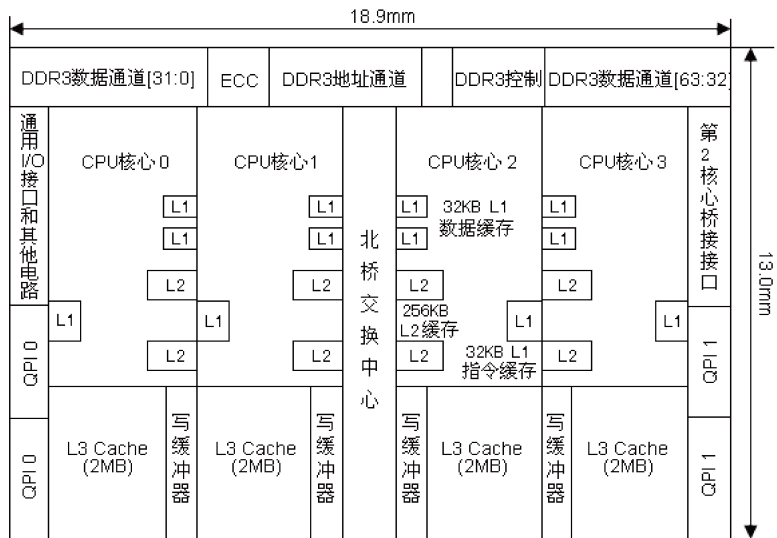


图 3-6 Intel Core i7 CPU 内核示意图

### 3.1.5 CPU 接口形式

CPU 是一个独立的产品，它需要通过接口与主板连接后才能工作。CPU 接口经过多年的发展，采用的插座形式有双列直插（DIP）插座、Socket 零插拔力插座、LGA（触点阵列封装）无引脚插座、BGA（球栅阵列封装）表面贴装等形式。不同类型的 CPU 接口，它们的插孔数量和形状都不相同，工作电压也不相同，不能互相混用，因此插座一般采用防呆设计。

从 80486 开始，CPU 开始使用一种“Socket”的零插拔力（ZIF）插座，它通过插座旁边杠杆的开合，可以将 CPU 很轻松地放入插座中，然后将压杆压回原处，利用插座本身结构产生的挤压力，将 CPU 的引脚与插座牢牢地接触，以消除 CPU 引脚与主板插座接触不良的问题。拆卸 CPU 时，只需要将压杆轻轻抬起，则压力解除，CPU 即可轻松取出，Socket 插座即使多次使用也不会造成磨损。Socket 插座大多根据 CPU 引脚的多少进行编号，如 Intel 公司早期使用的 Socket 370、Socket 423、Socket 478 等 CPU 插座；AMD 公司使用的 Socket 754、Socket 939、Socket 940 等插座，AMD 公司目前仍然在使用这类插座，如 940 针的 Socket AM2+、938 针的 Socket AM3 插座（如图 3-7 所示）。

Intel 公司 LGA 775、LGA 1156、LGA 1366 封装的 CPU 插座没有引脚，只有一排排整齐排列的金属圆点，因此 CPU 不能利用引脚进行固定，而需要一个安装扣架固定，使 CPU 可以正确压在 LGA 插座上的弹性触须上。LGA 封装原理与 BGA 封装相同，只不过 BGA 是用锡焊在主板上，而 LGA 插座可以随时解开扣架，更换芯片（如图 3-7 所示）。

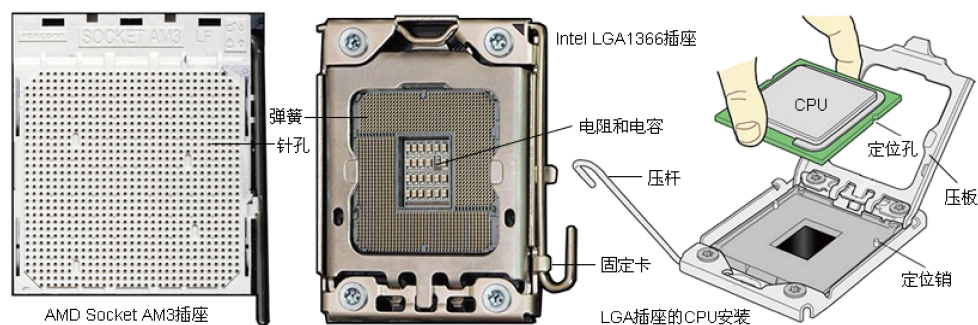


图 3-7 主板上的 CPU 插座形式与安装方法

AMD 公司的 CPU 产品采用了短针脚设计，Intel 公司则取消了 CPU 针脚。因为针脚有一定的电容性，会产生噪声干扰，加上针脚形同天线，非常容易接收干扰。而且针脚越长，噪声信号越大。虽然减短针脚可以降低噪声，但针脚太短又可能出现接触不良，而且增加了生产成本。LGA 无针脚封装技术虽然成本较高，但能解决干扰问题。

## 3.2 CPU 基本结构

### 3.2.1 CPU 系统结构

#### 1. CPU 工作过程

如图 3-8 所示，CPU 工作过程大致如下：指令和数据在执行前，首先要加载到内存或 CPU 内核的高速缓存（L1/L2/L3 Cache）中，这个过程称为缓存。CPU 根据指令指针（PC）寄存器指示的地址，从高速缓存或内存中获取指令；然后对分支指令进行预测工作，这个过程称为取指令（IF）。CPU 取到指令后，需要判断这条指令是什么类型的指令，需要执行什么操作，并负责把取出的指令译码为微操作（ $\mu$ OP）指令，这个过程称为译码（DEC）。指令译码后可以得到操作码和操作数地址，然后根据地址取操作数。然后需要对多条微操作指令分配计算所需要的资源（如寄存器、加法器等），这个过程称为指令控制（ICU）或指令分派。当操作数被取出来以后，计算单元（如 ALU）根据操作码的指示，就可以对操作数进行正确的计算了，指令的计算过程称为执行（EXE）。执行结束后，计算结果被写回到 CPU 内部的寄存器堆中，有时需要将计算结果写回到缓存和内存中，这个过程称为退出（Retire）或写回。到此为止，一条指令的整个执行过程就完成了。

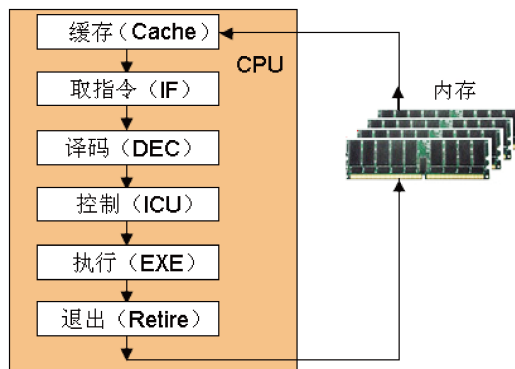


图 3-8 CPU 指令执行过程示意图

## 2. Core i7 CPU 系统结构

CPU 系统结构始终围绕着速度与兼容两个目标进行设计。改变 CPU 的系统结构有很多困难，主要是受到兼容性问题的困扰。一个全新的结构虽然有利于提高计算性能，简化 CPU 结构，但是必然会造成硬件和软件之间的不兼容现象。另外，新的结构必然导致 CPU 加工工艺的重大改变，而这将造成制造成本大幅增加。因此，在 CPU 设计中，往往在保证与以前指令系统兼容的基础上，改进 CPU 的系统结构。这种设计思想导致了 CPU 的结构越来越复杂。

如图 3-9 所示，Core i7 CPU 包括几十个系统单元。从体系结构层次看，CPU 的内部结构主要有缓存单元（Cache）、取指单元（IF）、译码单元（DEC）、控制单元（ICU）、执行单元（EXE）、退出单元（RU）等。

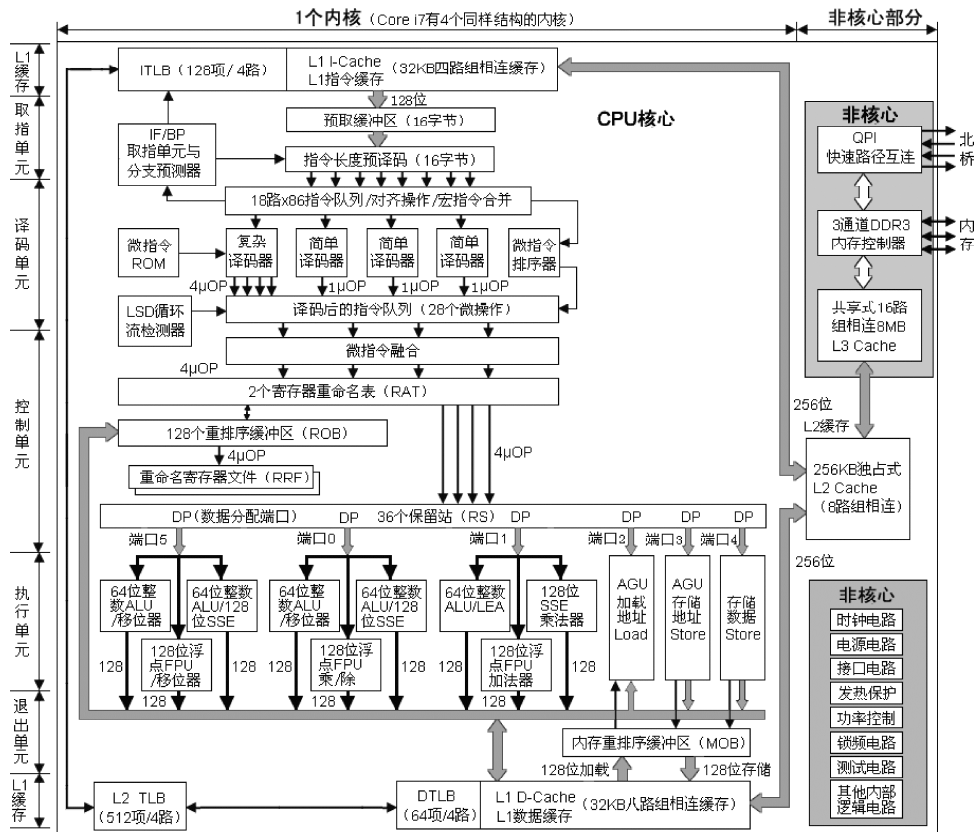


图 3-9 Intel Core i7 CPU 内核系统结构

## 3. Core i7 CPU 指令执行速度

如图 3-9 所示，Core i7 CPU 每个单核有 5 个 64 位整数算术逻辑运算单元（ALU），3 个 128 位的浮点处理单元（FPU）。CPU 中每个核心在最好的情况下，理论上每个时钟周期可以进行以下操作：取指令或数据 128 位/周期；译码 4 条 x86 指令（1 个复杂指令，3 个简单指令）/周期；发送 7 条微指令/周期；重排序和重命名 4 条微指令/周期；发送 6 条微指令到执行单元/周期；执行  $5 \times 64$  位 = 320 位整数运算/周期；或执行  $3 \times 128$  位 = 384 位浮点

运算/周期；完成并退出 4 条微指令（128 位）/周期。CPU 在 3.2GHz 频率下的峰值浮点性能为 51GFLOPS（双精度）或者 102GFLOPS（单精度）。

Core i7CPU 有 4 个物理核心，以下讨论单核下的基本结构和工作原理。

## 3.2.2 高速缓存单元

### 1. 存储器局部性原理

对大量程序运行情况的统计表明，在一个较短的时间内，指令的地址往往集中在内存地址空间的很小范围内。这种对局部范围的内存地址频繁访问，而对此范围以外的地址则访问比较少的现象，称为存储器局部性原理。局部性分为时间局部性和空间局部性，时间局部性指近期被访问的程序代码，很可能不久将再次被访问；空间局部性是指地址上相邻近的程序代码，可能会被连续地访问。

指令地址的分布一般是连续的，即程序往往重复使用它刚刚使用过的数据或指令。再加上循环程序段和子程序段需要重复执行多次，因此，对这些地址的访问就具有空间上集中的倾向。

数据分布的随机性较大，集中存放的倾向不如指令明显。但是程序中经常使用数组这种数据结构，它在内存单元中的分布也是相对集中的。

### 2. 高速缓存技术

根据存储器局部性原理，可以在内存与 CPU 寄存器之间设置一个高速（与计算单元速度同步）的和容量相对较小的存储器，将一部分马上需要执行的指令或数据，从内存复制到这个存储器中，供 CPU 在一段时间内使用。这个介于内存与 CPU 之间的高速存储器称为 Cache（高速缓冲存储器，来自法语 *caler*，原意是隐藏，发音为 *cash*）。CPU Cache 全部操作由硬件进行控制，对软件和程序员都是透明的。

Intel 公司从 80486 开始采用 Cache 技术来改善 CPU 性能，其他存储器与 Cache 之间的关系如图 3-10 所示。

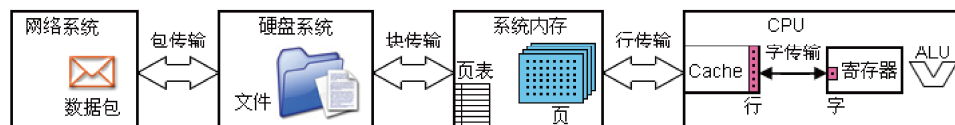


图 3-10 其他存储器与 Cache 之间的关系

### 3. 高速缓存的命中率

CPU 访问存储系统时，在存储系统中找到所需数据的概率称为命中率，命中率计算方法如下所示，命中率越接近于 1 越好。

$$\text{命中率} = \frac{\text{命中次数}}{\text{访问次数}} \quad (3-1)$$

CPU 访问存储系统时，通常先访问 Cache，由于 CPU 所需要的信息不会百分之百地在 Cache 中，这就存在一个命中率的问题。从理论上说，只要 Cache 的大小与内存的大小保

持适当比例，Cache 的命中率是相当高的。对于没有命中的指令或数据，CPU 只好再次访问内存，这时 CPU 将会浪费更多的时间。

为了保证 CPU 访问 Cache 时有较高的命中率，Cache 中的内容一般按一定的算法进行替换。较常用的算法有“最近最少使用算法”（LRU），它是将最近一段时间内最少被访问过的 Cache 数据行淘汰出局。目前 CPU 高速缓存的命中率可达到 95% 以上。

#### 4. Core i7 CPU 高速缓存结构

Core i7 CPU 高速缓存结构如图 3-11 所示。Intel 公司将 CPU 中的缓存分为一级缓存(L1 Cache)、二级缓存(L2 Cache)和三级缓存(L3 Cache)。在 L1 缓存中，分为数据缓存(D-Cache)和指令缓存(I-Cache)，两者可以同时被 CPU 访问，减少了争用缓存造成的冲突，提高了处理器效能。L1Cache 容量一般为 32KB(AMDL1Cache 为 64KB)；L2Cache 容量为 128KB~2MB；L3Cache 容量一般在 2~12MB。Cache 容量是提高 CPU 性能的关键。

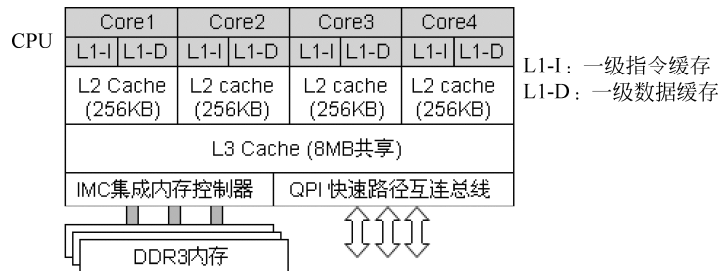


图 3-11 Intel Core i7 CPU 高速缓存结构

CPU 中 80% 的数据申请都可以在 L1 Cache 中命中，只有 20% 的数据申请需要另外查找；而这 20% 的数据申请中的 80%，又可以在 L2 Cache 中找到；在拥有 L3 Cache 的 CPU 中，只有大约 4% 左右的数据申请需要从内存中调用。可见，3 级缓存结构大大提高了 CPU 的运行效率。

#### 5. TLB 的基本功能

x86 CPU 的寻址方式非常复杂，Pentium 及以后的 CPU 采用虚拟地址寻址，虚拟地址=虚页号+页内偏移量。虚拟地址中的虚页号对应于物理内存中存放的“页表”（如图 3-12 所示）。页表存放在内存中，页表的数量与内存大小有关。CPU 每次获取数据都需要对内存进行两次访问。第一次是根据虚拟地址访问内存页表，获得访问数据的实页号；第二次是根据虚拟地址中的页内偏移访问数据的物理地址，这种访问模式对 CPU 性能影响很大。

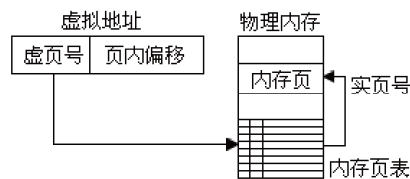


图 3-12 虚拟地址与物理地址的映像

为了减少 CPU 访问内存的次数，CPU 内部设计了 TLB（快表）。TLB（如图 3-13 所示）

是一个专用的高速缓存，用于存放近期经常使用的页表项，内容是内存中页表部分内容的副本。这样，CPU 进行地址转换时，可以在 TLB 中直接进行。只有偶尔在 TLB 不命中时，才需要访问内存中的页表。TLB 是将虚拟内存地址转换成物理内存地址的硬件单元。CPU 寻址时会优先在 TLB 中进行查询，这样 CPU 的性能就得到了提高。

### 3.2.3 取指令单元

#### 1. 指令预取单元结构

Intel Core i7 CPU 的指令预取单元 (IFU) 由 16 字节 (128 位) 的指令预取缓冲区、16 字节的指令长度预译码单元和指令分支预测器组成 (如图 3-13 所示)。

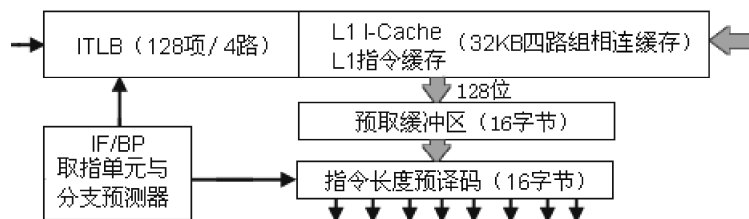


图 3-13 Intel Core i7 CPU 取指令单元结构

#### 2. 指令预取缓冲区

Core i7 处理器在每个时钟周期内，可以从 L1 I-Cache (指令缓存) 中取回一个长度为 16 字节 (128 位) 的指令块。如果指令的平均长度不超过 4 个字节，那么 Core i7 平均每个时钟周期可以处理 4 条指令。如果指令的平均长度为 5 字节，那么 Core i7 平均每个时钟周期最多能处理 3 条指令。这样，当指令长度过大时，就无法满足译码器的需求。如果指令预取单元一次能取回 32 字节的代码块，那么译码器就能保证每个时钟周期译码 5 条指令。

需要注意的是，处理器的很多指令长度有可能大于 5 个字节。在 SSE 扩展指令集中，一个简单的 SSE 指令可以在两个寄存器之间完成数据传输。在 64 位模式中，SSE 指令的长度经常会高达 7~9 个字节。

#### 3. 指令长度预译码器

x86 指令的长度是可变的，因此很难确定这些指令的长度。为了使指令长度不影响指令的执行速度，Core i7 处理器在正式译码前，会对指令长度进行预译码。指令长度预译码后，就可以区分出简单指令和复杂指令，并且将它们分派到不同的译码器。

#### 4. 指令分支预测器

在进行分支指令处理时，CPU 需要对执行指令进行预测，以避免出现译码中断。x86 分支指令包括无条件分支 (如 CALL 语句) 和有条件分支 (如 IF 语句)。无条件分支总是改变程序的流程。有条件分支则分为两种情况，当条件成立时，改变程序流程；当条件不成立时，按原有顺序执行下一条指令。应用程序中约有 10% 的语句是无条件分支，10%~20% 的语句是有条件分支。如果 CPU 中没有分支指令预测单元，当遇到 “if then else” 这

样的语句，就必须等待条件判断成立之后，才能继续执行计算，这会浪费不少时间。

分支指令预测的基本设计思想是：设立一个分支目标缓冲区（BTB），其中存放最近一次运行时，分支判断成功的信息（如指令地址、分支目标、指令指针等）。如果当前指令与分支目标缓冲区中某一条指令的地址相同，则确定该指令是分支指令，并预测成功，从分支目标缓冲区直接获得目标指令指针。反之，则顺序取指令。

分支指令预测技术在不改变指令系统的情况下，大幅度提高了 CPU 流水线的计算性能，缺点是实现技术复杂。

### 3.2.4 译码单元

由于 x86 指令集的指令长度、格式与寻址模式相当复杂，为了简化数据通道的设计，x86 处理器采用了将 x86 指令译码成一个或多个长度相同、格式固定、类似 RISC 指令形式的微指令（Micro-OP 或  $\mu\text{OP}$ ）。所以，目前 x86 处理器计算单元真正执行的指令是译码后的微指令，而不是编译后的 x86 指令。

#### 1. 译码单元结构

x86 指令译码单元的设计非常困难，增加译码虽然会大大增强处理器的译码能力，但是译码单元电路复杂，会提高 CPU 内核的复杂度和功耗。如图 3-14 所示，Core 处理器的译码单元设计有 4 个译码器，其中，3 个简单指令译码器，1 个复杂指令译码器，每个时钟周期可以生成 7 条微指令（ $\mu\text{OP}$ ）。

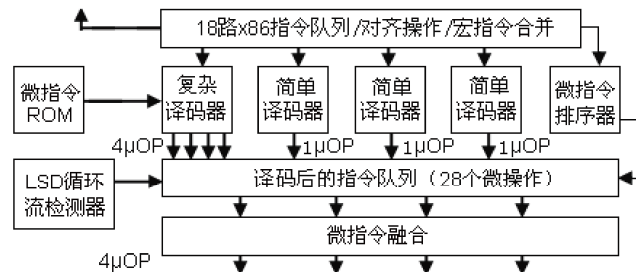


图 3-14 Intel Core i7 CPU 译码单元结构

指令译码分为硬件电路译码和微指令译码，从运算速度上看，硬件电路译码的速度比微指令译码要快，但是硬件译码电路复杂，而且扩展指令时，需要重新设计硬件译码电路。微指令译码速度慢，但是指令扩展容易实现。

#### 2. 简单指令译码器（SD）

简单译码器可以用来处理对应一条微指令的简单 x86 指令，例如，所有的 SSE 指令都可以用简单译码器处理，生成一条微指令。在实际应用中，往往采用硬件电路实现简单 x86 指令的译码。

#### 3. 复杂指令译码器（CD）

对于复杂的 x86 指令，往往采用微指令进行译码器设计，复杂译码器用来处理对应 4

条微指令的复杂 x86 指令。如果遇到非常复杂的指令，还可以借助旁边的微码 ROM 取得微指令序列。例如，向量指令是一种复杂的 x86 指令，它需要微码 ROM 和译码器共同完成译码工作。由于应用程序很少使用复杂的 x86 指令，因此复杂指令译码器对 CPU 的整体性能影响不大。

#### 4. 循环回路探测器 (LSD)

一旦译码器将 x86 指令译码成微操作指令后，就将微指令放入译码后的指令队列（微操作缓冲区），这时循环流检测器（LSD）就会对微指令进行循环预测。循环流检测器用于正确预测循环的结束，它记录下每个循环结束前所有的详细分支地址，当下一次同样的循环程序运行时，CPU 内核的 ROB（重排序缓冲区）和 RS（保留站）就可以准确快速地完成任

#### 5. 微指令 ROM

复杂译码器能将 x86 指令译码为 1~4 个微指令长度，而比这还要复杂的指令（如 15 个字节长度的指令），就需要由微指令 ROM 和复杂译码器共同处理。微指令控制的基本思想是：将复杂的 x86 指令编制成多条微指令，以简化控制操作；然后由若干微指令组成一段微程序，解释执行一条复杂的 x86 指令；微指令编制好后，事先存放在 ROM 控制存储器（CM）中，执行复杂 x86 指令（多于 4 条微指令）译码时，由微指令 ROM 和复杂译码器共同译码。由于微指令译码速度慢于硬件译码，因此对不常使用或复杂的指令才调用微指令译码。

### 3.2.5 控制单元

#### 1. 指令控制单元基本结构

Core 处理器的指令控制单元（ICU）采用乱序执行技术（OOO），如图 3-15 所示，主要部件包括寄存器重命名表（RAT）、重排序缓冲区（ROB）、重命名寄存器文件（RRF）和保留站（RS）等。

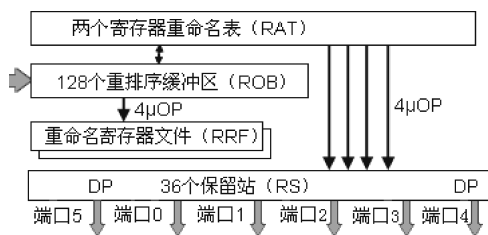


图 3-15 Intel Core i7 CPU 指令控制单元结构

#### 2. 乱序执行 (OOO) 技术

乱序执行指 CPU 允许多条指令，不按程序规定的顺序，分开发送给各相应电路单元进行处理，然后将处理结果重新排序。

例如，某一程序片段有5条指令，这时CPU乱序执行引擎将检查这些指令能否提前执行（相关性检查）；如果没有指令和数据相关，就根据各单元电路的空闲状态，将能够提前执行的指令立即发送给相应电路执行。各单元乱序执行完指令后，乱序执行引擎再将运算结果重新按原来程序指定的顺序排列。

在按序执行中，一旦遇到指令依赖的情况，流水线就会停滞，如果采用乱序执行，就可以跳到下一个非依赖指令并发布它。这样，执行单元就可以总是处于工作状态，把时间浪费减到最少。采用乱序执行技术的目的是为了CPU内部电路满负荷运转，提高CPU运行程序的速度。

### 3. 寄存器重命名（RAT）

在具有多个执行单元（如ALU）的CPU设计中，多个执行单元可以同时计算一些没有数据关联性的指令，从而总体提升计算效率。在乱序执行技术中，不同的指令可能会需要用到相同的通用寄存器（GPR），特别是指令需要改写该通用寄存器的情况下，为了让这些指令能并行计算，解决方法是对一些寄存器进行重命名，不同的指令可以通过具有相同名字，但实际不同的寄存器来解决。

Core i7的寄存器重命名表（RAT）指明每一个寄存器要么进入重排序缓冲区（ROB），要么进入重命名寄存器文件（RRF），并且ROB保存绝大多数最近的推测执行状态，而RRF则保存绝大多数最近执行的非推测状态。RAT分配给每一个微操作在ROB中一个目的寄存器，微指令读取它们ROB中的源操作数，并发送到保留站（RS）。

### 4. 重排序缓冲区（ROB）

经过寄存器重命名后，微指令将分发到ROB（重排序缓冲区）中，同时发送到RS（保留站），这个阶段称为分发。ROB的职责就是始终保持跟踪微指令的运行，并且控制它们的退出。ROB将乱序执行完毕的微指令，按照原始顺序重新排序，以保证所有的微指令在逻辑上实现正确的因果关系。

### 5. 保留站（RS）

RS（保留站）保存了所有等待执行的微指令，保留站中的微指令和数据可以直接进入ALU（算术逻辑运算单元）。RS的另一个作用是监听内部总线，保留站内是否有需要读取L1或L2缓存乃至内存的指令，或者需要等待其他指令结果的指令。

## 3.2.6 执行单元

### 1. 执行单元（EXE）结构

如图3-16所示，执行单元包括5个64位的整数算术逻辑运算单元（ALU），3个128位的浮点处理单元（FPU）和3个128位的SSE向量处理单元（其中两个与ALU共用）。其中3个浮点单元和3个SSE单元共享某些硬件资源。

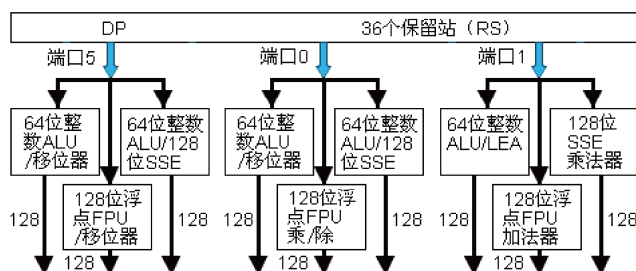


图 3-16 Intel Core i7 CPU 执行单元结构

## 2. 数据分配端口（DP）

为了配合 Core i7 的 36 个保留站（RS），Core i7 设计了 6 个数据分配端口（DP），每个端口有多个不同的单元，以执行不同的任务。然而在同一时间内，一个数据分配端口只能由一条微指令进入。因此也可以认为，Core i7 处理器有 6 个执行单元，在每个时钟周期内可以执行最多 6 个操作（或者说 6 条微指令）。

## 3. 整数执行单元（IEU）

Core i7 处理器的单核有 3 个 64 位 IEU（复杂整数执行单元，即 ALU），每个执行单元可以独立处理一个 64 位的整数数据；另外，Core i7 还有两个 SIU（简单整数处理单元，即 ALU）来快速运算较简单的任务。

对 x86 CPU 来说，Core CPU 首次实现了在 1 个时钟周期内，并行完成 5 条 64 位的整数运算。但是，由于 3 组整数执行单元使用了各自独立的数据分配端口，因此 Core i7 处理器的单核可以在一时钟周期内同时执行 3 组 64 位的整数运算。

## 4. 浮点处理执行单元（FPEU）

Core 处理器内核中包含兼容 IEEE 754 和 IEEE 854 标准的浮点处理单元，用于加速 x86 浮点运算。浮点运算需要很高的精度，计算量大，结构也异常复杂。在处理器中，浮点运算与整数运算的指令调度是完全分离的，并且它们的处理方式也完全不同。

Core i7 处理器的单核具有 3 个并行工作的浮点处理执行单元（FPEU），可以同时处理向量和标量的浮点数据，位于端口 1 的 FPEU-1 浮点处理执行单元负责加减等简单的运算，而端口 0 的 FPEU-2 浮点处理执行单元则负责乘除等运算，端口 5 的 FPEU-5 浮点处理执行单元则负责移位等运算。这样 Core i7 具备了在一周期内完成 3 条浮点指令的能力。

## 5. 向量执行单元（SSE）

向量计算是一种新型的浮点指令，MMX（64 位）、3DNow!（64 位）、SSE（128 位）都是向量指令集，向量指令也称为 SIMD（单指令多数据流）。

Core 处理器单核能够在单循环内完成 128 位向量运算，这样 Core 处理器可以将 128 位的大量 MUL（乘法）、ADD（加法）、Load（加载）、Store（存储）、COM（比较）、JMP（跳转）等指令，集成在一个周期中全部完成，使运算性能得到大幅提高。

### 3.2.7 退出单元

#### 1. 载入/存储单元（Load/Store）结构

运算需要用到数据，同样也会生成数据，这些数据的存取操作由 Load/Store（载入/存储）单元来完成（如图 3-17 所示）。Load（载入）单元的主要功能是将数据从存储器（内存或缓存）加载到运算单元的寄存器中（内存中的数据不变）；Store（存储）单元主要功能是将计算结果从寄存器写回存储器中（内存中的数据可能改变）。在 Core i7 处理器中，一个 Store 操作可分解成为两个微操作，一个用于计算地址，另外一个用于数据存储，这种方式可以提前预知 Store 操作的地址，初步解决了数据相关性问题的。

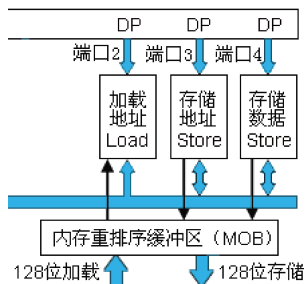


图 3-17 Core i7 CPU  
退出单元结构

在 CPU 中，Load/Store 操作（如 MOV、PUSH 等）十分频繁，约占所有指令的 1/3，因此它对系统性能的影响很大，特别是 x86 这样通用寄存器（GPR）较少的 CPU。Load 操作的发生频率比 Store 操作高，并且，Store 操作并不是影响系统性能的关键因素。因为在数据开始写入后，CPU 可以马上开始进行下面的工作，而不必等到写入操作完成。“缓存-内存”控制会负责将数据的整个部分写入到缓存，并复制到内存。

#### 2. 内存重排序缓冲区（MOB）

MOB 的主要功能是对指令乱序执行的结果，重新按原指令顺序进行排序。从 MOB 中移出一条指令就意味着这条指令执行完毕了，这个阶段称为 Retire（退出），相应地，MOB 往往也称为退出单元。

在一些超标量 CPU 设计中，退出阶段会将 MOB 的数据写入 L1 D-Cache（1 级数据缓存），而在另一些设计里，写入 L1 D-Cache 由另外的队列完成。例如，Core i7 处理器的这个操作就由 MOB（内存重排序缓冲区）来完成。

## 3.3 CPU 设计技术

### 3.3.1 x86 指令系统

#### 1. x86 基本指令集

Intel 公司于 1978 年发布了 8086 指令集。这些指令分为两部分，一部分为标准 8086 指令，另一部分为 8087 浮点处理指令，一共 166 条，这些指令奠定了 x86 指令集的基础。

x86 指令是一种复杂指令系统，而且没有什么规律。x86 指令长度为 1~15 字节不等，大部分指令在 5 个字节以下。从 Pentium Pro CPU 开始，Intel 公司将长度不同的 x86 指令，

在 CPU 内部译码成长度固定的 RISC（精简指令系统）指令，这种方法称为微指令或微操作（ $\mu$ OP）指令。

## 2. MMX 扩展指令集

1997 年，Intel 公司推出了 MMX（多媒体扩展）指令集，它包括 57 条多媒体指令。MMX 指令主要用于增强 CPU 对多媒体信息的处理能力，提高 CPU 处理 3D 图形、视频和音频信息的能力。在传统 8086 指令集中，无论多小的数据，一次只能处理一个数据，因此耗费时间较长。为了解决这一问题，在 MMX 技术中采用了单指令多数据（SIMD）技术，可使用一条指令对多个数据同时进行处理，它一次可以处理 64 位任意分割的数据。

## 3. SSE 扩展指令集

1999 年，Intel 公司在 Pentium III CPU 产品中推出了 SSE（数据流单指令序列扩展）指令集，Intel 和 AMD 公司希望用 SSE 指令集来取代 x87 浮点指令集。

SSE 指令集主要包含优化内存中连续数据块传输指令，提高 3D 图形运算效率的 SIMD 浮点运算指令，MMX 整数运算增强指令等。这些指令对图像处理、浮点运算、3D 运算、视频处理、音频处理等多媒体应用起到全面强化的作用。

## 4. x86-64 指令集

AMD 公司推出的 x86-64 指令集主要用于 x86 系列 CPU 进行 64 位计算。x86 是最成功的指令集，从目前来看，x86 指令集并不会很快淘汰，因为它是全球主流应用软件所采用的标准指令集，因此延续 x86 指令集的使用寿命是目前最经济的方式。

### 3.3.2 流水线技术

流水线技术是现代计算机系统设计中的一项关键技术。

#### 1. 流水线的基本设计方法

流水线设计的基本方法是：将一个重复的指令执行过程分解成为若干个子过程，而每个子过程都可以在专门设计的功能段上与其他子过程同时执行。每个子过程称为流水线的“级”或“段”。“级”的数量称为流水线的“深度”。各个功能段所需的时间应尽量相等，否则，时间长的功能段将成为流水线的瓶颈，会造成流水线的“堵塞”和“断流”，功能段的时间一般为一个时钟周期。

在 CPU 中，一条指令的执行过程可以分解为：取指令（IF）→指令译码（DEC）→取操作数（MEM）→执行计算（EXE）→写回结果（WB）。当指令获取单元读取了第 1 条指令后，马上进行第 2 条指令的获取操作，并不需要等待指令执行完成后再进行。CPU 的译码、执行等单元也是这样，这就形成了一条流水线作业系统。虽然流水线使指令的执行周期延长了，但能使 CPU 在每个时钟周期都有指令输出。流水线的不同结构如图 3-18 所示。早期在 CPU 内部设计过多条流水线（如图 3-18（e）所示），但由于制造复杂，目前很少采用这种设计方案。目前大部分 CPU 设计方案是采用一条流水线，但是设计多个功能处理单元（如图 3-18（f）所示）。

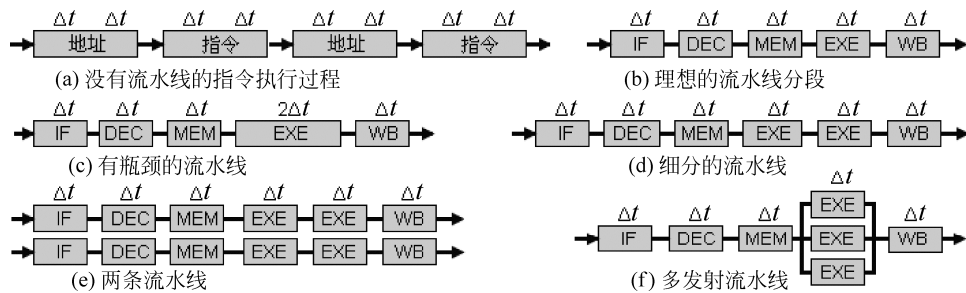


图 3-18 不同形式的流水线结构示意图

## 2. 理想的流水线

流水线设计的关键在于时序的合理安排,如果前级操作时间恰好等于后级的操作时间,设计最为简单,前级的输出直接汇入后级的输入即可。如果前级操作时间大于后级的操作时间,则需要对前级输出的数据适当缓存,才能汇入到后级输入端。如果前级操作时间恰好小于后级操作时间,则必须通过复制逻辑,将数据流分流,或者对前级数据采用存储后处理方式,否则会造成后级数据溢出。理想的流水线必须具备以下条件:

- (1) 所有指令都必须通过相同的流水段顺序流出。
- (2) 两个流水段之间不共享任何资源。
- (3) 通过所有流水段的操作和传输延时都相等。
- (4) 调度一个指令进入流水线后,不会对流水段中其他部件造成影响。

理想的流水线工作过程如图 3-19 所示。

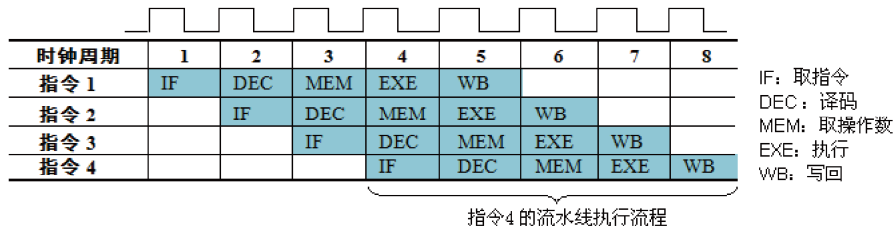


图 3-19 理想状态下的 CPU 流水线指令执行流程

流水段的时间是由最慢的流水段来决定的,因此,可以将几个操作合并成一个流水段,也可以将一个操作拆分成几个流水段,而不会影响流水线性能。

## 3. 流水线中的相关性

相关性是指一条指令的运行依赖于另一条指令的运行。流水线的性能会受到流水线中指令之间相关性的影响。流水线的相关性包括数据相关、资源相关和控制相关。

(1) 数据相关。流水线中的一条指令计算产生的结果,可能将在后面的指令中使用,这种情况称为数据相关。

**【例 3-1】** 某个程序指令队列如下:

A=100      (指令 1)  
B=200      (指令 2)

C=A+B (指令3)

从以上程序可见,指令1、指令2与指令3存在数据相关。在极端情况下,一条指令可以决定下一条指令的执行(如转移、中断等)。消除相关性的方法如下:一是保证在*i*+1阶段执行的指令和1~*i*阶段执行的指令无关;二是根据前一流水段的反馈来使用暂停指令或终止指令。如图3-20所示,在第5个时钟周期中,指令2正在取操作数B=200时,指令3同时进入了执行阶段,这时将产生错误的运算结果。因此指令3和指令4在第5个周期中,必须插入一个暂停周期(气泡),以消除数据相关性。暂停周期可以在程序编译时插入空操作指令(NOP)进行处理。

时钟周期 (t)	1	2	3	4	5	6	7	8	9
指令1: 取A	IF	DEC	MEM	EXE	WB				
指令2: 取B		IF	DEC	MEM	EXE	WB			
指令3: C=A+B			IF	DEC	气泡	MEM	EXE	WB	
指令4: 保存结果				IF	气泡	DEC	MEM	EXE	WB

IF: 取指令  
DEC: 译码  
EXE: 执行  
MEM: 取操作数  
WB: 写回

图3-20 程序代码C=A+B在流水线中执行的时空图

(2) 资源相关。流水线中的一条指令可能需要另一条指令正在使用的资源,这种情况称为资源相关。两条指令使用相同的寄存器或内存单元时,它们之间就会存在资源相关。例如,不能要求一个ALU既作地址计算,又作加法操作。

【例3-2】某个程序指令队列如下:

```
FOR A ≤ 100 (指令1)
A=A+1 (指令2)
B=B+A (指令3)
END
```

从以上程序可见,指令2与指令3存在资源相关。消除资源相关的方法有插入暂停周期(气泡)、重定向技术、设置相互独立的指令Cache和数据Cache等。

(3) 控制相关。分支指令会引起指令的控制相关性。如果一条指令是否执行依赖于另外一条分支指令,则称它与该分支指令存在控制相关。

【例3-3】某个程序指令队列如下:

```
IF p1 {
    s1
};
```

从以上程序可见,p1与s1存在控制相关。解决控制相关的基本方法是:与控制相关的指令不能移到分支指令之前,与控制无关的指令不能移到分支指令之后,减少或消除控制相关的方法是减少或消除分支指令。

#### 4. 数据冒险

在流水线的实际操作中,不可避免地会出现数据冒险问题。数据冒险是指在流水线中的数据尚未准备好之前,就对其进行访问操作。因此在硬件电路上,可以采用旁路技术消除这一问题。其次是流水线在运行中,可能会发生数据多取或漏取的现象,当发生这些异

常情况时，需要有专门的异常检测处理电路。

## 5. Core i7 CPU 流水线设计

Core i7 和 Core 2 CPU 都采用 14 级流水线设计，而此前 Pentium 4 CPU（Northwood 核心）的流水线为 20 级；Pentium D CPU（Prescott 核心）流水线为 31 级。从技术方面看，流水线越长，CPU 工作频率提升潜力越大；负面影响是一旦产生分支预测失败或者高速缓存取指令不能命中，CPU 就需要到内存中取指令，这时流水线必须清空（如图 3-21 所示），并重新执行流水线操作，因此耽误的延迟时间就会增加。例如，Pentium D CPU 发生分支预测失败或者缓存没有命中时，就会产生 39 个时钟周期的延迟。流水线越长，相关性和指令转移两大问题也越严重。因此，流水线并不是越长越好，找到一个速度与效率的平衡点才是最重要的。

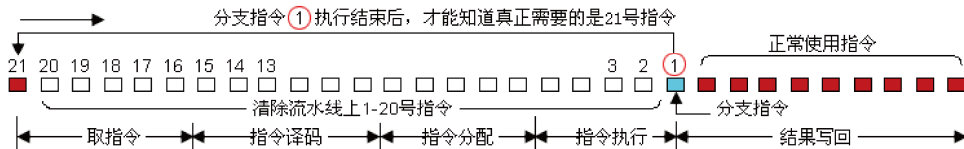


图 3-21 分支指令导致的流水线清空示意图

Intel 公司没有公布 Core i7 CPU 的流水线结构，可以参考 Intel Atom CPU 16 级流水线（如图 3-22 所示），大致了解 Core i7 CPU 的流水线结构。

时钟周期:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
流水线:	IF1	IF2	IF3	ID1	ID2	ID3	SC	IS	IRF	AG	DC1	DC2	EX1	FT1	FT2	IWB/DC1
	取指令			译码			分发	寄存器	存取数据缓存			执行	多线程处理		写回	

图 3-22 Intel Atom CPU 16 级流水线执行流程

## 6. 超标量技术

超标量技术是集成了多条流水线的 CPU（如图 3-18（e）所示），并且每个时钟周期可以完成一条以上的指令。80486 以下的 CPU 都属于单流水线结构，Pentium 及以上的 CPU 都具有超标量结构。流水线和超标量虽然可以提高指令运算速度，但是提高速度很难超过数倍。要想 10 倍、100 倍地提高运算速度，唯一解决的方法是设计多核 CPU，以及采用多 CPU 系统结构（如集群系统）。

### 3.3.3 多核 CPU 技术

为什么不用单核设计高性能 CPU 呢？这是因为功耗和发热问题限制了单核 CPU 不断提高性能的途径。如果通过提高 CPU 主频来提高 CPU 的性能，CPU 的功耗就会以指数级的速度急剧上升，很快就会触及所谓“频率高墙”的阻挡。CPU 温度和功耗的上升，使得 CPU 厂商不得不采用多核结构来提高 CPU 性能。

多核 CPU 带来了更强的并行处理能力，大大减少了 CPU 的发热和功耗。目前多核 CPU 有 2 核、4 核、8 核甚至过多内核的产品。2007 年，美国发布的“万亿级”80 核研究用 CPU 芯片，面积只有指甲盖大小，功耗为 62W。2011 年，Intel 公司实验型 CPU 最高主频达到

了 8.3GHz，而所需电能只有 83mW。Intel 公司还向公众展示了在 48 个奔腾级别核心整合为一体的实验型 CPU 技术。Intel 公司 48 核 CPU 测试如图 3-23 所示。

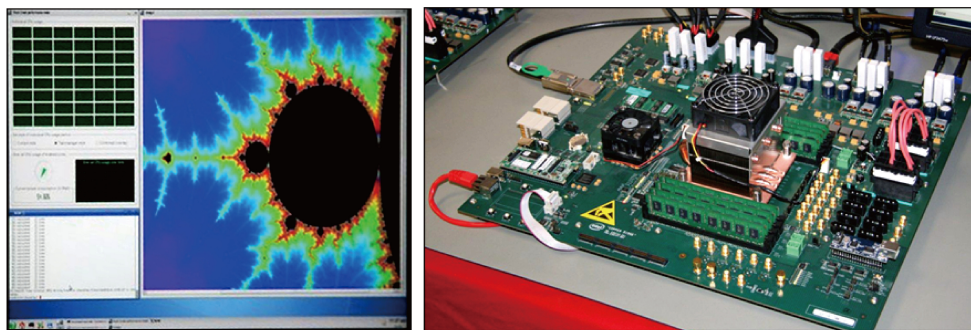


图 3-23 Intel 48 核 CPU 测试和主板

多核 CPU 的内核拥有独立的 L1 和 L2 缓存，共享 L3 缓存、内存子系统、中断子系统和外设。因此，系统设计师需要让每个内核独立访问某种资源，并确保资源不会被其他内核上的应用程序争抢。

多核的出现让计算机系统设计变得更加复杂。如运行在不同内核上的应用程序为了互相访问、相互协作，需要进行一些独特的设计。例如，高效进程之间的通信机制、共享内存的数据结构等。程序代码的迁移也是个问题，大多数软件开发商对单核 CPU 结构的程序设计进行了大量投资，如何使这些代码最大限度地利用多核资源，也是一个急需解决的问题。目前应用程序在从单核环境向多核系统的迁移过程中。

多核 CPU 需要软件的支持，只有在基于线程化的软件上，应用多核 CPU 才能发挥出应有的效能。目前大多数软件都是基于单线程的，多核 CPU 并不能为这些应用程序带来效率上的提高，因此多核 CPU 的最大问题是软件问题。

### 3.3.4 CPU 设计热功耗

#### 1. CPU 功耗

CPU 功耗（功率）是一项重要技术参数，根据电路基本原理，CPU 功耗如下：

$$\text{CPU 功耗} = F \times C \times V^2 \quad (3-2)$$

式中， $F$  为 CPU 内部晶体管的开关频率， $C$  为 CPU 内部电容负载， $V$  是 CPU 核心工作电压。可见要降低 CPU 的功耗，可以通过降低 CPU 运行频率、减少 CPU 工作电压和寄生电容来达到目的。目前台式计算机 CPU 的功耗为 50~130W。

#### 2. CPU 设计热功耗（TDP）

TDP（设计热功耗）是反映 CPU 热量释放的技术指标，它的含义是当 CPU 达到最大负载时释放出的热量，单位为 W。TDP 并不是 CPU 的实际功耗，显然 TDP 小于 CPU 的实际功耗。CPU 功耗是对主板提出的要求，要求主板能提供相应的电压和电流；而 TDP 是对散热系统提出的要求，要求散热系统能把 CPU 发出的热量散发掉。

Intel 公司 Coer i7 CPU 热功耗与 CPU 温度的关系如图 3-24 所示。

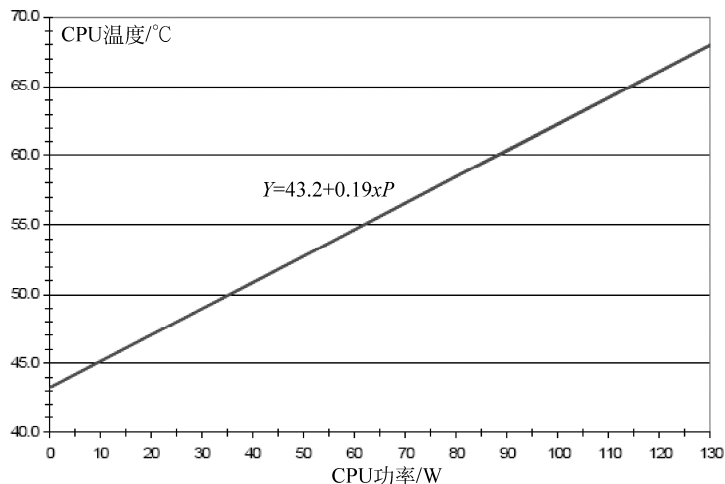


图 3-24 Intel Coer i7 CPU 功耗与温度的关系

TDP 值越小说明 CPU 发热量小，散热也越容易。对于笔记本电脑来说，电池的使用时间也越长。Intel 和 AMD 对 TDP 的定义并不完全相同。TDP 值不能完全反映 CPU 的实际发热量，因为目前 CPU 都设计有节能技术，实际发热量显然要受节能技术的影响。节能技术越有效，实际发热量越小。

### 3.3.5 CPU 节能技术

统计资料表明，当前主流配置的台式计算机平均工作功率大约为 200W，嵌入式系统平均工作功率为 30W 左右。这需要昂贵的封装、散热片以及冷却环境，并且 CPU 中的大电流会使产品寿命和可靠性降低。

#### 1. 动态功耗与静态功耗

CPU 中的功耗有两种：来自晶体管开关的动态功耗和来自漏电流的静态功耗。动态功耗又有：电容的充放电过程；MOS 晶体管中 PN 结同时打开时，形成的瞬间短路电流等。静态功耗有：MOS 晶体管扩散区和衬底形成的反偏电流；关断晶体管时通过栅极的电流等。CPU 芯片的漏电流会随温度而变化，所以芯片发热时，静态功耗会上升。

在 130nm 制程工艺以前，CPU 的功耗主要是动态功耗，静态功耗所占比例很小。在 90nm 制程工艺以后，由于线宽变小，漏电流问题严重。近几年，静态功耗问题逐渐被重视。Intel 公司发布的 45nm 的 Penryn 微结构 CPU，通过使用高  $k$  值电介质和金属栅极，来减轻静态功耗。也可以在 CPU 芯片空闲时，降低 CPU 频率和工作电压，来削弱漏电流的影响。但这种方式仍然无法完全解决漏电流引起的问题。尤其在芯片处于空闲时，大量的逻辑单元和复杂的时钟电路仍然需要消耗一定的电能。

#### 2. CPU 节能设计的基本方法

减少 CPU 功耗的理想方案是不同工作模式下采用不同的工作电压，但这会造成 CPU

设计太过于复杂。例如，这种方案需要考虑不同电压区域的隔离、开关及电压的恢复、触发器的状态、存储器中数据的恢复等问题。简单的设计方法是：按照高性能高电压，低性能低电压的原则进行设计，即使用多种时钟频率来降低芯片中部分单元的工作频率。

Core i7 CPU 增强了电源管理功能，CPU 中大部分功能单元在不使用时，可以进入睡眠状态以降低耗电。另外，Core i7 CPU 内部的多条总线为了提高传输带宽，加大了总线位宽，但是多数情况下并不需要如此大的传输位宽，因此这些总线平时只开放较低位宽，在没有信号传输时，也可以进入睡眠状态。

### 3.3.6 CPU 温度控制技术

Intel 公司从 Pentium 2 开始，在 CPU 内部设计了温度控制电路。这项技术的目的在于对 CPU 进行过热保护，避免 CPU 因为温度过高而引发火灾。

#### 1. CPU 温度状态控制

对 CPU 的测试表明，当 CPU 内核温度超过了某个警戒值时，CPU 将启动温度控制电路来降低 CPU 工作频率，防止温度进一步上升。CPU 温度控制状态如图 3-25 所示。

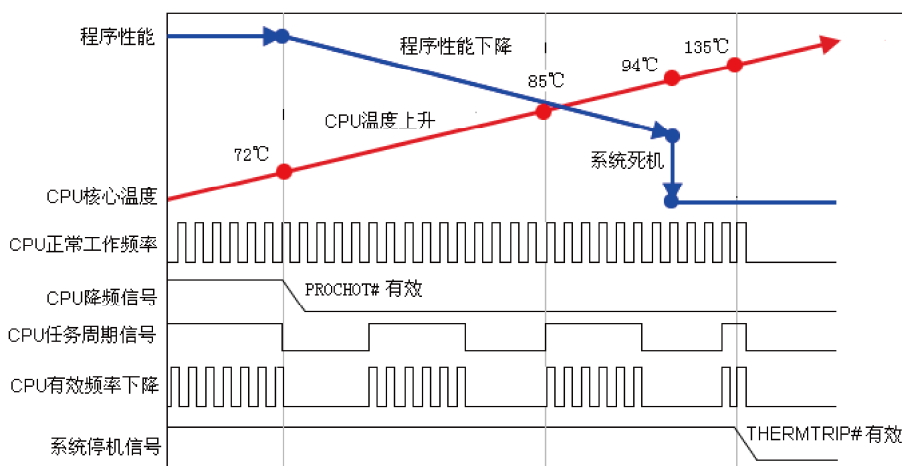


图 3-25 CPU 温度控制电路工作状态示意图

对 Pentium 4 系列 CPU 来说，CPU 的警戒温度值为  $72^{\circ}\text{C}$ ，极限温度为  $135^{\circ}\text{C}$ 。当 CPU 内核温度达到  $72^{\circ}\text{C}$  时，CPU 内部的温度控制系统向 CPU 发送 PROCHOT#(CPU 降频信号) 指令；CPU 内核的温度控制电路激活后，它会发出一个按比例（可以在 BIOS 中设置比例值）降低的 CPU 任务周期信号，强制 CPU 的工作频率按这个信号指定的周期运行，达到降低 CPU 工作频率的目的。当 CPU 温度继续升高到  $85^{\circ}\text{C}$  时，CPU 性能下降幅度将超过 50%。如果 CPU 温度继续升高到  $94^{\circ}\text{C}$ ，就会出现系统死机现象。当 CPU 温度继续升高到  $135^{\circ}\text{C}$  时，CPU 将自动发出关机指令（THERMTRIP#），从而关闭计算机系统。

目前普遍使用的酷睿 i 系列 CPU 产品，警戒温度和极限温度是多少呢？Intel 公司没有给出一个明确的说法。根据 Intel 公司的设计热功耗（TDP）规定，只要不超过  $T_j \text{ Max}$  温度就算是正常工作。例如，Intel Core i5 的  $T_j \text{ Max}$  温度是  $105^{\circ}\text{C}$ ，根据 Intel 公司的说法，

只要不超过 105℃ 都是正常工作范围。但是 105℃ 是否就是警戒温度尚不得知。

## 2. CPU 内部温度监控系统

Intel 公司早在 Pentium 4 CPU 中就集成了温度控制电路（TCC），Intel 公司将它命名为温度监控器（TM）。目前的 CPU 采用第 3 代温度监控系统，它的特点是：在 CPU 内核中集成了两套相互独立的热敏二极管。第一套热敏二极管监测 CPU 温度，并传输给主板上的温度监控电路，这套电路通过关闭系统来保护 CPU，不过只是在紧急情况下才会启用。第二套热敏二极管集成在 CPU 内核温度最高的部位，如 ALU（算术逻辑单元）附近。CPU 工作时，这两套热敏二极管的电阻和电流会随温度而变化。

各款 CPU 的警戒温度和极限温度值，由 CPU 制造商根据制造工艺和封装形式确定，并在技术白皮书中给出。为了防止用户自行设定而带来危险，Intel 将 CPU 的警戒温度和极限温度写入 CPU 内部的温控电路中，用户无法修改它们。

## 3.4 CPU 故障分析与处理

### 3.4.1 CPU 产品选择

#### 1. 根据应用需求选择 CPU 产品

对于不同需求的用户来说，选购 CPU 产品有不同的区别。

（1）普通应用。普通应用是指用户利用计算机进行：文字处理（如 Word），课堂教学（如 PPT），网络信息浏览（如 IE、QQ 等），玩小型网络在线游戏，标清视频（如 360P 视频）播放等应用。普通应用的主要用户群体是办公自动化人员、企业文员、计算机初学者等。普通应用对 CPU 的要求不是很高，CPU 选择的主要原则是经济实用。

（2）专业应用。专业应用是指用户利用计算机进行：计算机辅助设计（如 AutoCAD）、计算机平面图形处理（如 Photoshop）、小型商业数据库应用（如 MS SQL）、大型专业软件（如 C++ 程序开发）应用、玩 3D 游戏、高清视频（如 1080P 视频）播放等。专业应用的主要用户群体是企事业单位专业人员，他们对 CPU 的要求较高，CPU 选择的主要原则是高速低热。

（3）高级应用。高级应用是指用户利用计算机进行：专业视频编辑（如 Premiere），三维图形设计（如 3D Max），网络服务器（如 IIS）应用，大型工程设计等。主要用户群体是大型企事业单位、超级计算机 DIY（自己动手做）爱好者等。对高级应用来说，CPU 选择的原则是越快越好，一般选择最新、最快，也是最贵的 CPU 产品。

#### 2. Intel 与 AMD 公司产品类型选择

x86 系列 CPU 主要有 Intel 和 AMD 两家公司的产品，两家公司在 CPU 产品上的电气参数和机械参数都不相同，因此需要不同的主板进行配套。

Intel 公司是 CPU 市场的领导者，因此它的 CPU 产品在硬件上稳定性较好，在软件应

用上兼容性较好，但是同类产品在价格上高于 AMD 公司。

AMD 公司 CPU 产品的最大优势在于价格便宜，而且对 CPU 超频支持较好。它的缺点是外部设备的驱动程序不足，这在安装新推出的操作系统或非 Windows 操作系统时，会遇到不兼容等问题。

### 3. 盒装与散装 CPU 的选择

CPU 产品有盒装与散装两种类型。盒装与散装 CPU 没有本质区别，它们在质量和性能上是相同的，主要差别是质保时间的长短以及是否带散热风扇。一般而言，盒装 CPU 保修期要长一些，通常为三年，而且附带有一台散热风扇，散装 CPU 质保时间一般是一年，不带散热风扇。但是，散装 CPU 由于没有包装，很容易造成 CPU 的一些外部损伤。

## 3.4.2 CPU 超频方法

### 1. 为什么进行 CPU 超频

超频是人为地提高某个部件规定的工作频率，使它的性能得到大幅度提升。CPU 公司在同一系列的产品中，按 CPU 工作频率的高低进行价格划分，因此超频最明显的动机是从低价低性能的 CPU 中，获得与高价 CPU 相同的性能。

CPU 超频需要采用高质量的主板、性能高和质量好的内存条，另外还需要增强系统的散热能力（如 CPU 散热、北桥芯片散热、内存条散热等），这使得 CPU 超频带来经济上的好处会抵消殆尽。况且 CPU 超频后，随之而来会出现一些系统稳定性方面的问题。因此，CPU 超频仅仅是计算机爱好者的一种乐趣而已。

### 2. CPU 超频的可行性

一条 CPU 生产线投资达三十多亿美元，而 CPU 公司有数千个不同型号的产品，不可能为每个型号的产品建立或调整一条生产线。为了降低 CPU 的设计和生产成本，大部分同一系列不同型号的 CPU，在设计中采用了同一系统结构，在制造上也采用同一生产线，在加工中采用同一制程工艺，甚至同一批次生产。最后通过 CPU 检测，将性能稳定的 CPU 标记为高频率型号，性能不太稳定的 CPU 标记为低频率型号。如果同一批次的 CPU 产品性能都很稳定，则人为地限定 CPU 的性能（锁频），使 CPU 在正常状态下，只能工作在某个规定的频率上。CPU 内部的 Cache 对 CPU 性能的影响也很大，通常 CPU 生产厂商也会对用户可使用的 Cache 容量进行人为分割或锁定。

人为频率锁定或 Cache 分割的方式，可以通过设备在硬件电路上进行自动处理；也可以利用 CPU 内部的微程序进行软件锁定，而且软件处理方式更加灵活方便。

CPU 工作频率的人为限定，为 CPU 超频提供了良好的基础。

### 3. 外频的基本概念

“外频”是一个复杂而且容易混淆的概念。如图 3-26 所示，计算机中的集成电路芯片和总线均按规定的时钟频率工作，而时钟频率由外部时钟电路（晶振+时钟芯片）按指定的标准（如 Intel CK505）提供，这是“外频”这一概念的基本含义。也就是说，外频是指从

外部输入到某个部件的时钟频率。

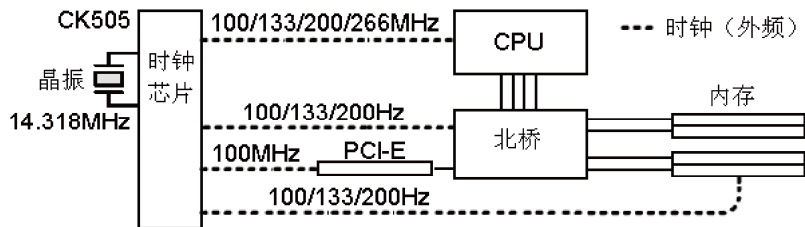


图 3-26 计算机系统 CK505 时钟频率标准（部分）

CK505 时钟标准为不同部件提供了不同的外频，对一些重要的部件，还提供了数个可供选择的的标准外频。例如，CPU 可供选择的的标准外频有 100/133/200/266MHz。在支持 CPU 超频的主板中，可以对标准外频进行修改，如将外频修改为 125MHz 或其他值。

#### 4. 倍频的基本概念

计算机中不同部件的工作速度不同，有些部件的工作速度与外频相同，例如，PCI 总线的外频为 33MHz，它工作的时钟频率也为 33MHz。也有些部件的内部工作频率大大高于外频。例如，CPU 内部工作频率远远高于外频，这就需要对外频进行倍频。CPU 内部工作频率（或 CPU 主频）与外频之间的相对比例关系称为倍频（或倍频系数）。

CPU 工作频率与外频和倍频的关系如下：

$$\text{工作频率 (MHz)} = \text{外频 (MHz)} \times \text{倍频} \quad (3-3)$$

根据以上公式，可以反向推算出 CPU 的外频。如 CPU 主频为 3.2GHz，倍频为 24 时，外频为  $3200\text{MHz} \div 24 = 133\text{MHz}$ 。

#### 5. CPU 超频对硬件的要求

CPU 超频对计算机的部分硬件设备有特殊要求：一是要有足够超频潜力的 CPU，二是支持 CPU 超频的主板，三是性能强大和可靠性高的内存条，四是有较强散热能力的设备（风扇、热管、水冷等），以及其他部件。

Intel 公司自 1998 年以来的 CPU，倍频是锁定不能改变的。在部分 CPU 中（如 AMD CPU），倍频是“封顶锁定”的，也就是可以改变倍频到更低的数字，但不能超过规定的最高倍频。在极少数 CPU 中，倍频是完全放开的，这类 CPU 是超频极品。提高或降低倍频进行 CPU 超频是最简单的超频方法，这种方法只影响到 CPU 的速度，而不会影响计算机其他部件。如果 CPU 的倍频已经锁定，就只能通过改变外频进行超频，而改变外频进行超频时，可能会带来各种各样的问题。

#### 6. 超频的方法与步骤

常见的超频方法是通过 BIOS 进行设置。计算机启动时按 Delete 键（部分计算机按 F1、F2、Esc 键），就可以进入 BIOS 设置菜单。如果主板支持超频功能，则在 BIOS 中有专门的超频设置项目。每个主板厂商对 CPU 超频的设置方法都不尽相同，如图 3-27 所示，技嘉公司的部分主板采用了图形界面 BIOS，它的 CPU 超频设置在 N.I.T.菜单中。



图 3-27 技嘉 X79 系列主板 CPU 超频 BIOS 参数设置

【例 3-4】某 Intel Core i7 CPU 的工作频率为 3.6GHz，外频为 100MHz，倍频为 36（没有锁频）。如图 3-27 所示，下面以技嘉 X79 系列主板的 BIOS 超频设置进行说明。

(1) 没有超频前，CPU 工作频率=100MHz（外频）×36（倍频）=3.6GHz。

(2) 开机启动，进入 BIOS 超频设置菜单（N.I.T.）。将 CPU 外频超频系数提高到 1.25，这时 CPU 外频=100MHz×1.25=125MHz。如果 CPU 倍频已经锁定，则 CPU 此时的工作频率=125MHz×36=4.5GHz。这项工作要逐步进行，外频超频系数一次不能提高太多。

(3) 如果 CPU 倍频没有锁定（Intel 产品极少），则可以在 BIOS 中，继续将倍频提高到 38，此时 CPU 工作频率=125MHz×38=4.75GHz。

(4) 内存频率设置。外频为 100MHz 时，原内存倍频为 21.33，内存传输频率=100MHz×21.33=2.133GHz；当外频调高到 125MHz 后，就会使内存的传输频率同步提升到 125MHz×21.33=2.666GHz，这会使内存工作在不稳定状态；因此必须将内存倍频减小为 16，这样内存传输频率=125MHz×16=2000MHz。

(5) Intel 酷睿 i 系列 CPU 具有自动超频功能（睿频技术），CPU 会自动根据程序运行情况改变 CPU 工作频率。为了保证系统的稳定性，必须在 BIOS 中关闭“睿频”功能。

(6) 保存 BIOS 设置参数，重启计算机进行超频稳定性测试。如果发生过热、死机、重启等故障，则需要调整 CPU 电压，加大散热力度，降低外频等工作。

以上是一个反复调整的过程，不能指望一次超频成功。

## 7. CPU 工作电压对超频的影响

超频时，如果不论采用多好的散热器都不能提高 CPU 工作频率时，可能是 CPU 没有获得足够的电压。这时必须在 BIOS 设置中提高 CPU 的工作电压。

提高 CPU 工作电压会导致 CPU 工作不稳定，甚至导致系统重新启动。每个 CPU 都有厂家推荐的电压设置，可以在厂家的网站上找到它们。CPU 电压的最大值由出厂时决定，并且无法更改，因此不要超过厂家推荐的最大工作电压。提高 CPU 工作电压后，发热量会大幅度增长，这时必须加强 CPU 的散热功能。

## 8. CPU 超频的危害

(1) 系统发热。CPU 超频时，它会产生更多的热量，如果散热不充分，就有可能造成

系统崩溃。不过一般的发热并不会烧坏计算机，例如，CPU 本身有发热保护功能，而系统随机重启是最常见的过热征兆。

(2) 工作不稳定。超频使得某些部件（如主板、内存、集成显卡、硬盘等）运行在高于规定的技术参数（如时钟周期、电压、功率、温度等）下，长时间工作在这种状态下时，会导致系统不稳定，产生经常性死机、蓝屏、重启等问题。

(3) 可能缩短计算机部件的使用寿命。

### 3.4.3 CPU 发热故障处理

#### 1. CPU 发热的原因

(1) CPU 超频。一些计算机爱好者喜欢进行“CPU 超频”工作，如果散热不好，超频会造成 CPU 发热严重。

(2) 散热装置不良。CPU 的工作电压越高和运行速度越高，所产生的热量也越大，因此必须使用品质良好的散热装置，来降低 CPU 的表面温度。

(3) 机箱散热风道不良。散热片和散热风扇的体积越大，散热效果越好；安装散热装置后，应留有充分的散热空间及散热风道。

#### 2. 检查机箱中散热装置是否正常

(1) 检查 CPU 散热片和风扇是否被灰尘堵死了，或者风扇轴里边油泥太多时，会导致风扇散热性能下降。清理风扇表面和散热片上的灰尘；对于风扇轴里的油泥，必须拆开风扇用纯酒精清洗，然后再加油。

(2) 查看 CPU 散热风扇的电源线是否连接好，用手拨动风扇，看旋转是否灵活。

(3) 检查机箱的风道是否合理，把机箱里面乱七八糟的线都绑在一起固定住。

(4) 加电启动系统，查看 CPU 风扇是否正常。如果风扇不转或转动速度低，会导致 CPU 散热不良，必要时更换散热效果更好的 CPU 风扇。

(5) 观察 CPU 引脚是否有氧化现象（表面发黑），这种情况多发生在南方潮湿多雨季节。可用酒精棉球将氧化部位擦干净，吹干后再安装到 CPU 插座。

#### 3. 检查 BIOS 中 CPU 参数设置是否正确

(1) 检测 CPU 电压设置是否正确。如果 CPU 工作电压过高，会使 CPU 工作时发热而死机；如果工作电压太低，CPU 也不能正常工作。

(2) 检查 CPU 频率设置。CPU 的外频设置过高会出现死机现象；如果外频设置过低，会使系统运行速度太慢。

(3) 检查芯片组设置。检查 CPU 与主板芯片组是否匹配，与内存条的型号或速度是否匹配，与外部设备接口的速度是否匹配等。

#### 4. 笔记本电脑发热处理

笔记本电脑的散热器由内置的温度传感器控制，散热风扇一般分为三个档次的风力：空闲，负载中等，负载高。当笔记本处于空闲状态时，散热风扇一般不启动，这时笔记本

很安静；当笔记本达到中等负载，系统温度在 50~55℃时，笔记本散热风扇启动，这时散热风扇声音不大；当笔记本处于高负载，系统温度在 60~70℃时，散热风扇就会全速运行，这时散热风扇噪声很明显。

如果笔记本发热严重，可以购买一个笔记本散热板，将笔记本放置在散热板上使用，这样可以明显缓解笔记本温度过热的的问题。

### 3.4.4 CPU 负载 100%故障处理

#### 1. CPU 占用率过高故障处理

(1) 防杀毒软件造成 CPU 占用率很高。目前的杀毒软件都加入了对网页、插件、邮件的随机监控，这无疑增大了 CPU 的负担。因此，应尽量使用最少的监控服务或者关闭杀毒软件的文件监控功能。

(2) 驱动程序没有经过认证，造成对 CPU 占用率达到 100%。大量测试版的驱动程序在网络上泛滥，这些程序没有经过微软公司的认证，很容易造成设备与操作系统的冲突，而导致 CPU 占用率过高。尤其是显卡驱动程序要特别注意，建议使用微软公司认证的或由产品厂商官方发布的驱动程序，并且严格核对型号和版本。有时可以适当升级驱动程序，但是最新的驱动程序不一定是最好的。

(3) 计算机病毒、木马程序造成 CPU 占用率 100%。大量计算机病毒在系统内部迅速复制，造成了 CPU 占用资源居高不下。用可靠的杀毒软件彻底清理内存和硬盘，经常性地升级杀毒软件和防火墙，加强防范意识。

(4) 打开多个在线视频网页时，CPU 占用率达到 100%。为了方便观看在线视频，用户往往同时打开多个视频网页，一边观看视频，一边等待另一网页进行视频数据缓冲。网络在线视频数据数据量很大，需要不断进行数据传输和数据解包，特别消耗 CPU 资源。因此，不要一次打开过多的视频网页。

(5) 某些常用软件 CPU 占用达到 100%。一些常用软件如浏览器 IE、压缩软件 WinRAR、视频编辑软件、图形处理软件等，本身占用了很高的 CPU 资源，要减少同时使用这些软件，或者用其他同类软件代替。

#### 2. CPU 常见故障分析

CPU 故障的现象主要是死机和速度降低，造成 CPU 故障的主要原因是散热不良。CPU 常见故障如表 3-4 所示。

表 3-4 CPU 常见故障现象

故障现象	故障主要原因分析
死机	CPU 超频；CPU 散热不良；CPU 工作电压过低；电源供电不足等
运行速度降低	CPU 温度过高引起自动降频等
重新启动	CPU 温度过高等

### 3.4.5 CPU 故障维修案例

**【例 3-5】** 开启“睿频”功能，导致 CPU 温度增加。

(1) 故障现象。计算机配置为，CPU Core i5，双核 4 线程；操作系统为 Windows 7 的 64 位版，在 Windows 7 操作系统中安装了虚拟机 VMware 软件，在虚拟机中安装了 Windows XP SP3 操作系统。在 Windows 7 下玩大型全屏游戏时，经常会出现屏幕闪烁现象，CPU 温度为 60 多摄氏度。在 VMware 虚拟机里玩同一大游戏时，CPU 温度达到了 92℃。

(2) 故障处理。虚拟机非常消耗系统资源，而在虚拟机里运行大型软件时，会导致系统性能不足。由于酷睿 i5 采用睿频技术，CPU 这时会自动超频，加速运行，因此导致 CPU 发热增加。在 BIOS 中关闭“睿频”功能，这时 CPU 温度最高只有 70℃了。

(3) 经验总结。不同程序都满载运行（CPU 使用率均为 100%）时，它们的发热量不一定相同。例如，运行 Intel Burn Test（CPU 稳定性测试软件）和进行视频压缩（压片）时，CPU 使用率都达到了 100%，但是压片时 CPU 的温度要高于测试程序；同样，进行 Furmark（显卡测试软件）的温度比玩游戏高。在虚拟机中运行某个软件时，酷睿 i5 会使用到专用的虚拟机指令（VT-x），这些虚拟机的电路非常复杂，会导致 CPU 发热更大。

**【例 3-6】** CPU 散热风扇设计问题，导致运行软件温度过高。

(1) 故障现象。新买的机器，散热没有问题，空载时用 EVEREST 软件检测的温度是：主板 25℃左右；CPU 开机不运行任何软件时，表面温度 64℃左右，核心温度 15℃左右；硬盘 25℃左右；显卡 50℃左右。玩游戏时，CPU 最高达到 80℃左右，运行 QQ 软件时，CPU 为 70℃左右。

(2) 故障处理。打开机箱，开机后触摸 CPU 铝散热片，感觉温度不高。分析 CPU 表面温度高，核心温度低，可能是风扇对着散热片吹，将热量吹到了 CPU 表面，导致 CPU 表面温度升高。然后尝试降低散热风扇的转速，发现 CPU 表面温度下降，而核心温度升高，说明散热风扇有问题。更换一个竖立横吹的散热风扇，故障消除。

(3) 经验总结。散热风扇的选择和安装非常重要，往下吹的风扇容易对 CPU 表面散热不利，竖立横吹的散热风扇容易导致将热风吹到内存条或北桥芯片；大风量风扇噪声过高等。散热片安装时，一是要保证与 CPU 表面接触良好，二是风扇的震动会导致接触不良。

**【例 3-7】** 旧笔记本计算机发热严重，CPU 利用率高。

(1) 故障现象。三年前购买的一台高配置笔记本电脑，4GB 内存。平时主要用于上网浏览网页，或者看在线视频。但是看视频时笔记本发热很厉害，感觉视频画面也卡得非常厉害，而且经常停顿。有时为了方便，开两个视频页面，一个看，一个先缓冲。这时候机器几乎要崩溃了，风扇噪声非常大，发热非常厉害。打开任务管理器，发现 CPU 使用率在 70%以上。

(2) 故障处理。按道理三年前高配置的笔记本电脑，看在线视频应该不成问题。分析故障原因有：笔记本散热风扇灰尘严重？CPU 与风扇散热片之间的硅胶干涸？感染了计算机病毒？下载软件时被强制安装过恶意软件？

笔记本没有重启，说明发热还没有达到极限，硅脂不一定要换。没有拆机，用强力吸尘器对准笔记本出风口，控制好距离，开吸尘器抽出灰尘，结果散热效果明显。

CPU 使用率过高，检查到底是哪个进程所致。打开任务管理器，看到有两个浏览器进

程,检测发现主要是浏览器进程占用了较高的CPU使用率;但打开任务管理器后,可以看到CPU使用率会从70%左右降低到30%~40%;为什么打开进程管理器后CPU使用率会下降呢?说明有可能是病毒或恶意软件所致。重新安装操作系统,故障排除。

(3) 经验总结。笔记本计算机在使用过程中要注意散热良好。

## 习 题

- 3-1 Intel公司的CPU产品有哪些类型?
- 3-2 简要说明CPU工作过程和主要部件。
- 3-3 简要说明存储器局部性原理。
- 3-4 画出CPU温度升高与程序性能的关系图,并说明典型温度点发生的现象。
- 3-5 一个4核3.2GHz的Core i7 CPU,时钟周期是多少?CPU性能如何?
- 3-6 讨论可以用软件的方法设计CPU吗。
- 3-7 讨论CPU引脚为什么越来越多。
- 3-8 讨论CPU超频的理论依据。
- 3-9 写一篇课程论文,论述CPU系统结构设计方面的技术。
- 3-10 利用CPU-Z等软件测试CPU基本技术参数。