

$$1. (1) \lim_{F \rightarrow 1} \frac{1}{1-F+\frac{F}{N}} = N$$

实际意义是：当整个系统都被改善时，系统的加速比就是改进获得的提升倍率 N ，无论系统改善多少部分，其改进获得的提升倍率，总不会大于 N ，实际改善中不能只一味提高 F ，还要提高 N

$$(2) \lim_{N \rightarrow \infty} \frac{1}{1-F+\frac{F}{N}} = \frac{1}{1-F}$$

实际意义：无论被改进的部分提升倍率有多高，整个系统的提升倍率总不会超过 $\frac{1}{1-F}$ ，所以不能一味地提高改进部分的性能，还要提高改进部分的比例，这样才能有效提高计算机性能

2、假设需要 N 个处理器核心，则程序的并行运行部分的运行能力提升 N 倍

$$\text{若加速比} = 5 = \frac{1}{1-0.9+\frac{0.9}{N}} \Rightarrow N = 9$$

∴ 需要 9 个处理器核心可以让程序获得 5 个加速比

$$F = 0.9 \text{ 确定当 } N \rightarrow \infty \text{ 时 } \frac{1}{1-F+\frac{F}{N}} \rightarrow \frac{1}{1-F} = 10 < 15$$

∴ 不可能获得 15 个加速比

3. (1) 若选择整型运算则： $F = 10\%$ $N = 3$

$$\therefore S_1 = \frac{1}{1-0.1+\frac{0.1}{3}} = \frac{15}{14}$$

若选择浮点运算则： $F = 60\%$ $N = 5$

$$\therefore S_2 = \frac{1}{1-0.6+\frac{0.6}{5}} = \frac{25}{13}$$

若选择内存访问, 则 $F=5\%$ $N=20$

$$\therefore S_3 = \frac{1}{1.005 + \frac{0.05}{20}} = \frac{400}{381}$$

S_2 最大, \therefore 选择浮点运算可获得最大整体加速比

(2) 实际的性能优化要综合考虑可优化部分所占比例 F 和优化幅度 N , 要选择优化 F 与 N 都较高的功能, 优化幅度很高但占比很小的功能也不是最优选

4. (1) N 个核会多至 $N\%$ 的通信开销.

\therefore 可优化比例为 $(M-N)\%$

可优化部分的提升倍率为 N

$$\therefore \text{总加速比 } S_{\text{overall}} = \frac{1}{1 - (M-N)\% + \frac{(M-N)\%}{N}}$$

(2) $M=80$

$$\text{则 } S_{\text{overall}} = \frac{1}{1 - \frac{(80-N)}{100} + \frac{80-N}{100N}} = \frac{1}{\frac{19}{100} + \frac{N}{100} + \frac{4}{5N}}$$

当 $\frac{N}{100} + \frac{4}{5N}$ 取最小值时 S_{overall} 取最大值

而 $\frac{N}{100} + \frac{4}{5N} \geq 2 \frac{1}{5\sqrt{5}}$ 当且仅当 $N = \sqrt{80}$ 取等号

又 $8 < \sqrt{80} < 9$ 且 $\frac{8}{100} + \frac{4}{5 \times 8} = 0.18 > \frac{9}{100} + \frac{4}{45} = 0.089$

\therefore 当 $N=9$ 时取到最佳加速比

7.

(1) 动态功耗

在 CPU 工作的时候,组成它的逻辑门状态实际是一直在发生着改变的,此时可以将逻辑门内部的晶体管视为小电容,电容的充放电就引起电能流动,此时就产生了 CPU 的动态功耗。这部分功耗的大小与 CPU 的运行状态有关。我们知道,主频是 CPU 最主要的规格之一,它指的是晶体管在一秒内的开关次数,晶体管开关次数越高,频率也就会更高,但总的充放电次数也同时增加了,于是晶体管的耗电量也就随之增加。

(2) 静态功耗

在栅级的下方有一层很薄的氧化层,它起到绝缘的作用。在把晶体管做小的过程中(也就是制程进步的过程中),这个绝缘层的厚度越来越薄,由于量子效应,部分电荷载流子将穿过这薄薄的绝缘层,出现漏电现象。这种漏电现象将增加晶体管的耗电量,虽然单个晶体管的漏电量相当微小,但是当晶体管的数量达到上亿甚至是数十亿级别之后,整个的漏电量就不容小觑了。

除了漏电以外,还有一部分静态功耗是晶体管出现导通情况引起的。在逻辑门状态发生改变的时候,组成它的部分晶体管可能会出现同时导通的情况,也就是短路的情况,此时会产生短路功耗,其大小与逻辑门的大小相关。

(3) 时钟功耗

时钟是惟一在所有时间都充放电的信号,而且很多情况下引起不必要的门的翻转,从而产生损耗。

解决方法:

(1) 采用门控时钟。

门控时钟包括门控逻辑模块时钟和门控寄存器时钟。门控逻辑模块时钟对时钟网络进行划分,如果在当前的时钟周期内,系统没有用到某些逻辑模块,则暂时切断这些模块的时钟信号,从而明显地降低开关功耗。门控寄存器时钟的原理是当寄存器保持数据时,关闭寄存器时钟,以降低功耗。

(2) 动态电压调节

动态功耗与工作电压正相关,功耗将随着电压降低而降低,因此降低工作电压是降低功耗的有力措施。但是,仅仅降低工作电压会导致传播延迟加大,执行时间变长。然而,系统负载是随时间变化的,因此并不需要微处理器所有时刻都保持高性能。动态电压调节 DVS 技术降低功耗的主要思路是根据芯片工作状态改变功耗管理模式,从而在保证性能的基础上降低功耗。

(3) 并行结构与流水线技术

并行结构可以通过牺牲面积来降低功耗,并行设计后,由于有多个模块同时工作,提高了吞吐能力,可以把每个模块的速度降低为原来的 $1/n$,从而使功耗降低

流水线技术本质上也是一种并行。把某一功能模块分成 n 个阶段进行流水作业,每个阶段由一个子模块来完成,若工作频率不变,对某个模块的速度要求仅为原来的 $1/n$,这样也可以使得功耗降低。

(4) 低功耗单元库

设计低功耗单元库是降低功耗的一个重要方法,包括调整单元尺寸、改进电路结构和版图设计。可以根据负载电容和电路延时的需要选择不同尺寸的电路来实现,这样会导致不同的功耗,因此可以根据需要设计不同尺寸的单元。同时,为常用的单元选择低功耗的实现结构,如触发器、锁存器和数据选择器等。

(5) 低功耗状态机编码

状态机编码对信号的活动性具有重要影响,通过合理选择状态机状态的编码方法,减少状态切换时电路的翻转,可以降低状态机的功耗。其原则是:对于频繁切换的相邻状态,尽量采用相邻编码。

(6) Cache 的低功耗设计

作为现代微处理器中的重要部件,Cache 的功耗约占整个芯片功耗的 30%~60%,因此设计高性能、低功耗的 Cache 结构,对降低微处理器的功耗有明显作用。Cache 低功耗设计的关键在于降低失效率,减少不必要的操作。通常用来降低 Cache 功耗的方法有以下两种:一种是从存储器的结构出发,设计低功耗的存储器;另一种是通过减少对 Cache 的访问次数来降低功耗。

8.

量子计算机 (quantum computer) 是一类遵循量子力学规律进行高速数学和逻辑运算、存储及处理量子信息的物理装置。当某个装置处理和计算的是量子信息,运行的是量子算法时,它就是量子计算机。量子计算机通过量子力学规律以实现数学和逻辑运算,处理和储存信息。它以量子态为记忆单元和信息储存形式,以量子动力学演化为信息传递与加工基础的量子通讯与量子计算,在量子计算机中其硬件的各种元件的尺寸达到原子或分子的量级。量子计算机是一个物理系统,它能存储和处理用量子比特表示的信息。

优势:

(1) 量子计算机运行速度较快、处置信息能力较强、计算能力强。与一般计算机比较起来,信息处理量愈多,对于量子计算机实施运算也就愈加有利,也就更能确保运算具备精准性。

(2) 传统的计算机通常会受到病毒的攻击,直接导致电脑瘫痪,还会导致个人信息被窃取,但是量子计算机由于具有不可克隆的量子原理这些问题不会存在,在用户使用量子计算机时能够放心地上网,不用害怕个人信息泄露。

劣势:

(1) 量子消相干:量子计算的相干性是量子并行运算的精髓,但在实际情况下,量子比特会受到外界环境的作用与影响,从而产生量子纠缠。量子相干性极易受到量子纠缠的干扰,导致量子相干性降低,也就是所谓的消相干现象。

(2) 量子纠缠。量子作为最小的颗粒,遵守量子纠缠规律。即使在空间上,量子之间可能是分开的,但是量子间的相互影响是无法避免的。而量子纠缠会影响量子的相干性,进而影响计算机的性能。

(3) 量子不可克隆。

量子不可克隆性,是指任何未知的量子态不存在复制的过程,既然要保持量子态不变,则不存在量子的测量,也就无法实现复制。对于量子计算机来说,无法实现经典计算机的纠错应用以及复制功能。