

9. 解: 对^{性能}不同微处理器架构进行性能对比与分析: 计算机最本质的定义是“完成一个任务所需要的时间”, 执行性能与时间成反比。在考虑相同指令集处理器时, 通常用 CPI 或 IPC 作为衡量标准。

在实际操作中, 通常运行测试程序, 通过数值定量反映性能, 主要影响因素有时钟周期, CPI 和程序执行指令数。常见的微处理器测试程序有: microbench, bp-microbench, Lmbench, STREAM, Coremark, Coremark-pro, Unixbench, Dhrystone 等。

Dhrystone 由 Reinhold P. Weicker 于 1984 年提出, 主要测试处理器的整数运算和逻辑运算的性能。Dhrystone 测试方法很简单, 即单位时间内跑了几次 Dhrystone 程序, 指标单位为

DMIPS/MHz。MIPS 是 million instruction per second 的缩写, 每秒处理百万级机器

语言指令数; DMIPS 中 D 是 Dhrystone 的缩写, 通常把在 VAX-11/780 上的测试结果

175 Dhrystones/s 定义为 1 DMIPS, 因此其他平台测得的结果除以 175, 故 DMIPS 其实是一个相对值。

CoreMark 跑分是通过运行 C 语言代码得出来的分数, 主要包含如下运行规则: 列举寻找并排序, 数学矩阵操作 (普通矩阵运算), 状态机 (确定输入流中是否包含有效数字), CRC (循环冗余校验)。测试在配置参数的组合下运行测试程序次数, 单位 CoreMark/MHz。

10. 解: 层次化设计特点体现如下:

① 处理器架构: 现代计算机通常采用多级流水线和多级缓存的结构。多级流水线将指令执行分若干阶段, 使处理器可以同时执行多条指令;

② 存储器架构: 存储器层次包括寄存器、高速缓存、主存、辅助存储器等结构;

③ 操作系统: 现代操作系统也采用了分层结构, 如 Linux 系统的内核空间和用户空间;

④ 网络协议: 网络协议也采用了分层结构, 如 TCP/IP 协议中的应用层、传输层、网络层和链路层;

⑤ 计算机系统: 从低到高物理层、总线层、操作系统层、应用程序层、用户层。

层次化设计提高了系统性能、可靠性和可维护性。通过分层设计, 各个层次之间的接口清晰明确, 可以更容易地进行系统扩展或维护。同时分层可以减少系统的复杂度, 提高系统可靠性和稳定性。

附加题2: A New Golden Age for Computer Architecture 在回顾历史的同时,提出了随着计算机领域的迅猛发展,硬件的性能和功能不断增强,计算机架构却面临着越来越多的挑战。摩尔定律、Dennard Scaling 定律的终结,还有 Amdahl 定律的限制,计算机架构在功耗、性能、可靠性、安全性等方面有更高的要求。

在此基础上, Hennessy 提出了自己的看法。在此之前,面对各个领域复杂问题,人们通常寻求统一理论和方法进行解决,计算机架构同样如此,但这不可避免的会带来效率的降低。Hennessy 列举指令及功耗浪费的例子,从而提出硬件与软件协同设计,设计高级、特定于领域的语言和体系结构,根据特定领域的固有特点进行专门化优化,以通用性换取定向的高性能,从而极大地解放了架构工程师的创造力,开辟出计算机架构的新黄金时代。