

《计算机组成与系统结构课程设计》指导书

张运杰

电子与信息工程系计算机应用教研室

2010-11-18

一、课程设计目的

1. 融会贯通计算机组成原理课程和计算机系统结构课程的内容，通过知识的综合运用，加深对计算机系统各模块的工作原理及相互联系的认识，特别是对硬布线控制器的认识。
2. 学习运用 ISP（在系统编程）技术进行设计和调试的基本步骤和方法，熟悉集成开发软件中设计、模拟调试工具的使用，体会 ISP 技术相对于传统开发技术的优点。
3. 培养科学研究的独立工作能力，取得工程设计与组装调试的实践经验。

二、课程设计要求

1. 设计流水方案时，牢记设计的目的是提高系统性能。没有性能改善、为流水而流水的方案是毫无意义的。
2. 实验系统的时序发生器将一个微指令周期分为 T1 至 T4 四段，原则上，本次实验只利用四段划分，即仍使用 T1—T4 作为时序脉冲。
3. 设计时，应充分考虑控制信号的综合和化简，出厂时的模型机提供了这方面的某些化简实例，但还可以进一步化简。

三、课程设计环境

- （1）TEC—4 计算机组成原理实验仪一台
- （2）双踪示波器一台
- （3）直流万用表一只
- （4）逻辑测试笔一支
- （5）集成电路若干片，取决于设计方案。

四、课程设计内容

1. 在微程序控制器实验的基础上，设计一台微程序控制的指令级标量流水模型计算机。
根据设计图纸，在通用实验台上进行组装，并调试成功。
2. 在组装调试成功的基础上，整理出设计图纸和其他文件。包括的文件是
 - a) 总框图（数据通路图）；
 - b) 微程序控制器逻辑图；
 - c) 微程序流程图；
 - d) 微程序代码表；
 - e) 元件排列图；
 - f) 设计说明书；
 - g) 调试小结。

五、课程设计步骤

1. 总体设计

指令级标量流水,是指把机器指令的解析过程分解为取指、译码、访存、执行、写回等子过程,各子过程以流水方式运行。考虑到复杂度,在实际设计中,可以分解得不那么细。例如可只分为取指、执行、写回三个子过程。

2. 流水微程序控制器

(1) 微指令格式

在本实验中仍采用水平型微指令格式,这是由于采用垂直型微指令来控制并行度较高的流水线将会困难重重。微命令编码仍然可以用直接表示法,后继地址用断定方式。

严格地说,没有中断控制,相应的控制信号可以省略,因此微指令字长可以缩短。为了兼容模型机方案,仍需保留这些控制信号,以减少接线的困难。

(2) 微程序控制器

支持流水并未对控制器的硬件结构提出更高的要求。微程序控制器的逻辑结构与模型机的差异的只是顺序控制部分,即微程序转移逻辑电路,原因是微程序需要重新设计,重新分配微地址。新设计的微程序转移逻辑电路,可以用一片 ispLSI1032 实现,也可用中小规模的数字器件实现。

(3) 时空图

如果是要从头设计一套流水系统,时空图应在设计数据通路之前确定,因为具体的流水线是跟时空图紧密相联的。本实验是在已有的数据通路的基础上(已知该数据通路支持流水),通过改进控制来提升系统性能,因此将时空图放在控制器部分进行设计。

(4) 微程序设计

控制器的强大与否,对机器性能发挥出多少,就取决于如何设计控制流程,并转化为微程序。微程序设计包括了横向设计和纵向设计。要实现流水控制,纵向设计仍然重要,而横向设计也同样需要仔细考虑。因为流水线中要求尽量多的并行操作,以便充分利用硬件资源,减少闲置。横向设计中就应把不冲突的、可以同时实现的控制,放在同一微指令中。纵向设计中,考虑控制顺序要连同并发控制一起考虑,尤其是出现冲突时,需要前后错开。此外就是注意在机器指令发生转移(无条件、有条件或中断)的地方,要丢弃已取的指令,重新做一次取指操作。

3. 组装与调试

对微程序控制器流水方案设计好了以后,形成初步的设计文件,然后按照设计文件进行组装与调试。在调试过程中,往往会发现设计中的问题,需要修改设计,在根据修改后的设计进行调试,直到完全成功为止。调试成功后,整理出最后的设计文件。