

3.10.1 理解指针	192	4.4 流水线的通用原理	282
3.10.2 应用: 使用 GDB 调试器	193	4.4.1 计算流水线	282
3.10.3 内存越界引用和缓冲区 溢出	194	4.4.2 流水线操作的详细说明	284
3.10.4 对抗缓冲区溢出攻击	198	4.4.3 流水线的局限性	284
3.10.5 支持变长栈帧	201	4.4.4 带反馈的流水线系统	287
3.11 浮点代码	204	4.5 Y86-64 的流水线实现	288
3.11.1 浮点传送和转换操作	205	4.5.1 SEQ+: 重新安排计算 阶段	288
3.11.2 过程中的浮点代码	209	4.5.2 插入流水线寄存器	289
3.11.3 浮点运算操作	210	4.5.3 对信号进行重新排列和 标号	292
3.11.4 定义和使用浮点常数	212	4.5.4 预测下一个 PC	293
3.11.5 在浮点代码中使用位级 操作	212	4.5.5 流水线冒险	295
3.11.6 浮点比较操作	213	4.5.6 异常处理	306
3.11.7 对浮点代码的观察结论	215	4.5.7 PIPE 各阶段的实现	308
3.12 小结	216	4.5.8 流水线控制逻辑	314
参考文献说明	216	4.5.9 性能分析	322
家庭作业	216	4.5.10 未完成的工作	323
练习题答案	226	4.6 小结	325
第 4 章 处理器体系结构	243	参考文献说明	326
4.1 Y86-64 指令集体系结构	245	家庭作业	327
4.1.1 程序员可见的状态	245	练习题答案	331
4.1.2 Y86-64 指令	245	第 5 章 优化程序性能	341
4.1.3 指令编码	246	5.1 优化编译器的能力和局限性	342
4.1.4 Y86-64 异常	250	5.2 表示程序性能	345
4.1.5 Y86-64 程序	251	5.3 程序示例	347
4.1.6 一些 Y86-64 指令的详情	255	5.4 消除循环的低效率	350
4.2 逻辑设计和硬件控制语言 HCL	256	5.5 减少过程调用	353
4.2.1 逻辑门	257	5.6 消除不必要的内存引用	354
4.2.2 组合电路和 HCL 布尔 表达式	257	5.7 理解现代处理器	357
4.2.3 字级的组合电路和 HCL 整数表达式	258	5.7.1 整体操作	357
4.2.4 集合关系	261	5.7.2 功能单元的性能	361
4.2.5 存储器和时钟	262	5.7.3 处理器操作的抽象模型	362
4.3 Y86-64 的顺序实现	264	5.8 循环展开	366
4.3.1 将处理组织成阶段	264	5.9 提高并行性	369
4.3.2 SEQ 硬件结构	272	5.9.1 多个累积变量	370
4.3.3 SEQ 的时序	274	5.9.2 重新结合变换	373
4.3.4 SEQ 阶段的实现	277	5.10 优化合并代码的结果小结	377
		5.11 一些限制因素	378
		5.11.1 寄存器溢出	378