

给定指令系统的处理器设计

根据计算机组成原理课程所学的知识，设计一个给定指令系统的处理器，深刻理解处理器结构和计算机系统的整体工作原理。

1. 处理器功能及指令系统定义

本实验的任务是设计一个简单的 RISC 处理器，该处理器是在给定的指令集下构建的，支持十条指令。假定主存可以在一个时钟周期内完成一次存取操作，而且可以和 CPU 同步工作。系统使用一个主存单元：指令读取和数据访问都使用同一组存储器。

处理器的指令字长为 16 位，包含 8 个 8 位通用寄存器 R0~R7，1 个 16 位的指令寄存器 IR 和 1 个 16 位的程序计数器 PC。取指令时，可以直接从主存中提取 16 位的指令信息，而进行数据访问时，与主存进行 8 位的数据交换。处理器的地址总线宽度是 16 位，数据总线宽度也是 16 位，无论是取指还是数据访问，都使用同一组数据总线，只是数据信息的宽度不同。

处理器所支持的指令包括 LDA, STA, MOV, MVI, ADD, SUB, JZ, JMP, IN, OUT。其中仅有 LDA 和 STA 是访存指令，所有的存储器访问都通过这两条指令完成；ADD 和 SUB 是运算指令，MOV 和 MVI 是传数指令，他们都在处理器内部完成；JZ 是跳转指令，根据寄存器的内容进行绝对跳转；JMP 是无条件转移指令；IN 和 OUT 是输入输出指令，所有 I/O 端口与 CPU 之间的通信都由 IN 和 OUT 指令完成。

1) 非访存指令

(1) 加法指令 ADD Ri, Rj

该指令在执行阶段需完成两个寄存器内容相加，结果送回寄存器的操作，具体为： $R_i + R_j \rightarrow R_i$

(2) 减法指令 SUB Ri, Rj

该指令在执行阶段需完成两个寄存器内容相减，结果送回寄存器的操作，具体为： $R_i - R_j \rightarrow R_i$

(3) 寄存器传送指令 MOV R_i, R_j

该指令在执行阶段只完成数据信息从寄存器 R_j 向寄存器 R_i 传送的操作，具体为： $R_j \rightarrow R_i$

(4) 立即数传送指令 MVI R_i, X

该指令在执行阶段只完成指令中的 8 位立即数 X 向寄存器 R_i 传送的操作，具体为： $X \rightarrow R_i$

2) 访存指令

采用扩充寻址的方式支持访存。

扩充寻址的定义：

(1) 存数指令 STA R_i, X

该指令在执行阶段需将寄存器 R_i 的内容存于主存单元中，对应的地址由 8 位形式地址 X 经扩充寻址生成，**R7 充当扩充寻址寄存器**，即主存实际地址记为 $R7//X$ ，具体操作为： $R_i \rightarrow [R7//X]$

(2) 取数指令 LDA R_i, X

该指令在执行阶段需将主存单元中的内容存于寄存器 R_i ，对应的地址由 8 位形式地址 X 经扩充寻址生成，**R7 充当扩充寻址寄存器**，即主存实际地址记为 $R7//X$ ，具体操作为： $[R7//X] \rightarrow R_i$

3. 转移类指令

(1) 条件转移（零则转）指令 JZ R_i, X

该指令根据寄存器 R_i 的内容决定下一条指令的地址，若寄存器内容为零，则 8 位形式地址 X 经寄存器 $R7$ 扩充寻址后形成有效地址 $R7//X$ ，送至 PC，否则程

序按原顺序执行。具体操作为：if ($R_i = 0$) then $[R7//X] \rightarrow PC$

(2) 无条件转移指令 JMP X

该指令改变下一条指令的地址，指令码中的 8 位形式地址 X 经寄存器 R7 扩充寻址后形成有效地址 $R7//X$ ，送至 PC，记为： $[R7//X] \rightarrow PC$

4. I/O 指令（选做，加分项 1）

(1) 输入指令 IN R_i , PORT

该指令完成从 I/O 端口到 CPU 的信息传送，指令码中的端口号 PORT 为端口地址，传送的是端口中的信息，送至 R_i ，记为： $[PORT] \rightarrow R_i$

(2) 输出指令 OUT R_i , PORT

该指令完成从 CPU 到 I/O 端口的信息传送，指令码中的端口号 PORT 为端口地址，记为： $R_i \rightarrow [PORT]$

2. 设计要求

要求根据以上给定的指令系统设计 RISC 处理器，处理器工作流程按取指、译码、运算、访存、写回五个阶段进行（或自行设计）。控制器设计方法可选（组合逻辑设计或微程序设计方法）

需完成的环节包括：

- 1) 指令格式设计；
- 2) 微操作的定义；
- 3) 节拍的划分；
- 4) 处理器结构设计框图及功能描述；
- 5) 如采用组合逻辑设计，列出操作时间表，画出每个控制信号的逻辑图；

- 6) 如采用微程序设计，写出每条机器指令对应的微指令序列，确定微指令字长和微指令格式，编写微指令码点；
- 7) 加分项 2：用 Verilog 实现该 CPU，并仿真验证其功能。

3. 其他要求及时间安排

1) 独立完成，严禁抄袭

2) 提交纸质版，统一 A4 纸，左侧装订。正文 5 号字宋体，西文用 Times New Roman，1.2 倍行距。要求排版规范，图表规范。

3) 提交时间：11 月 9 日（周三）课堂上提交。