**实验二 单周期十条指令 CPU**

**一、实验目的**

通过设计并实现支持 10 条指令的 CPU，进一步理解和掌握 CPU 设计的基本原理和过

程。

**二、实验内容**

设计和实现一个支持如下十条指令的单周期 CPU。

**非访存指令**

清除累加器指令 CLA

累加器取反指令 COM

算术右移一位指令 SHR：将累加器 ACC 中的数右移一位，结果放回 ACC

循环左移一位指令 CSL：对累加器中的数据进行操作

停机指令 STP

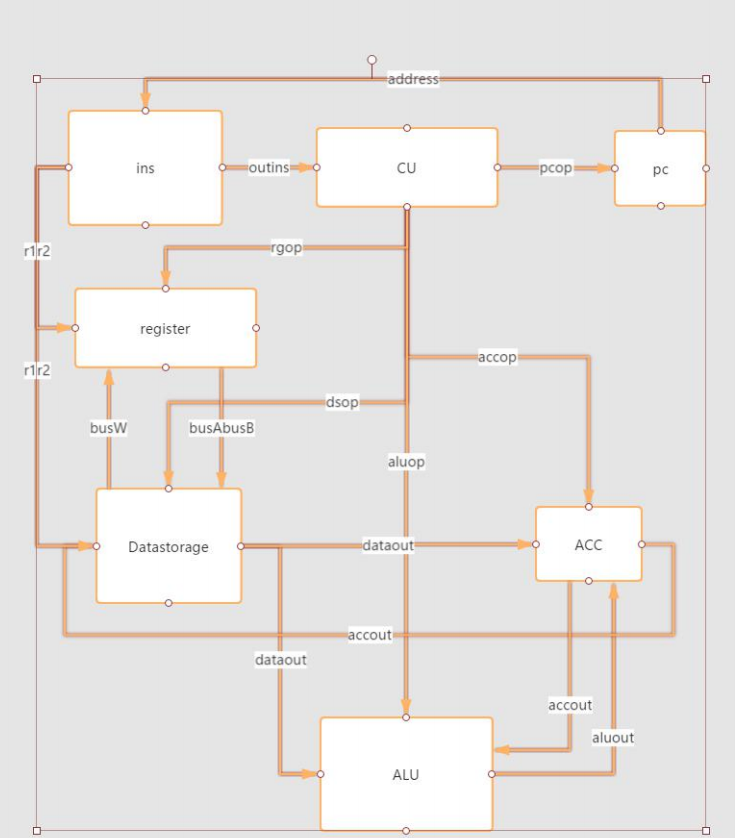
**访存指令**

加法指令 ADD X：[X] + [ACC] –〉ACC，X 为存储器地址，直接寻址

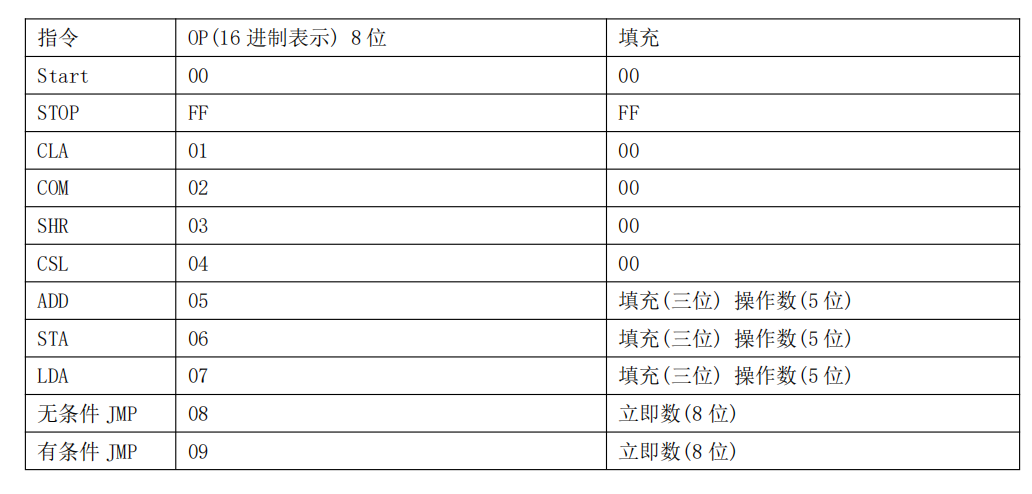
存数指令 STA X，采用直接寻址方式

取数指令 LDA X，采用直接寻址

1. **实验原理**



图一：cpu 原理图指令



**四、实验步骤**

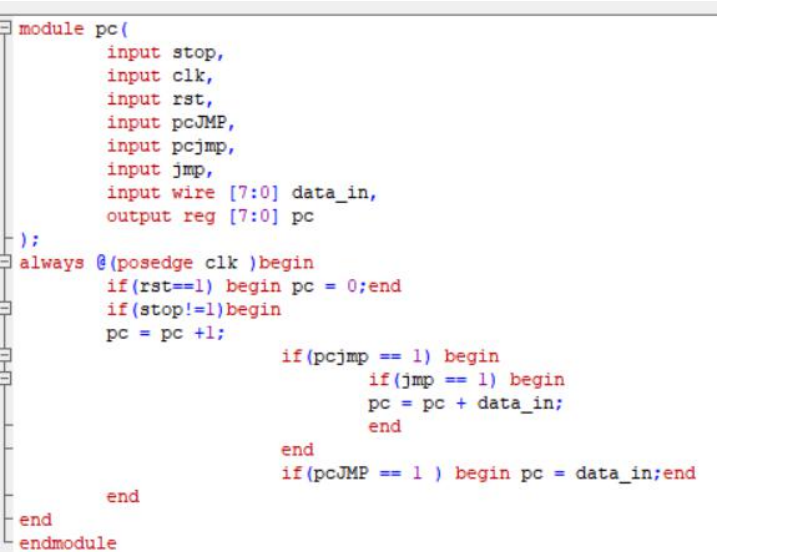
**4.1 CPU 各模块 Verilog 实现**

**1) PC 模块**



表 4.1 PC 功能描述

Verilog 关键代码：



**2) 指令存储器模块**

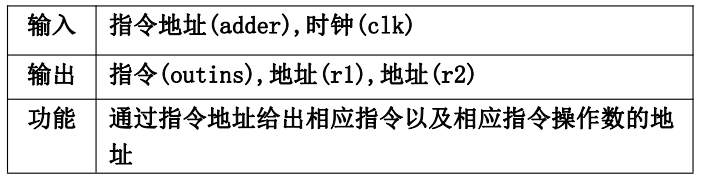
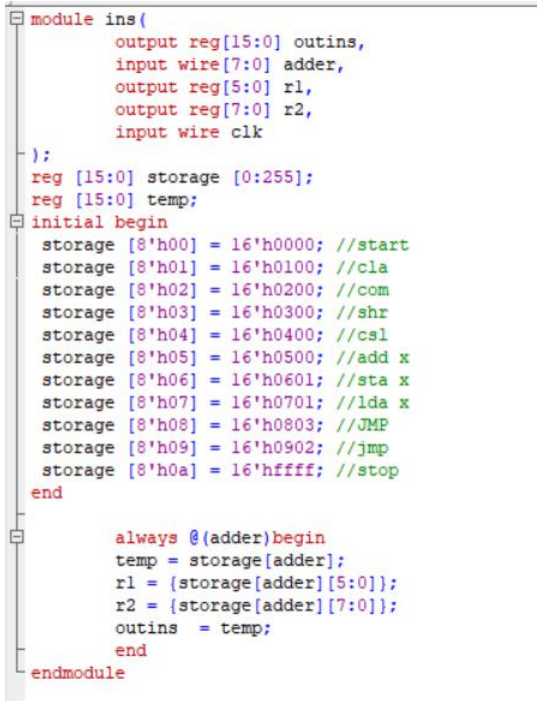


表 4.2 指令存储器模块功能描述

Verilog 关键代码：



**3) 寄存器堆**

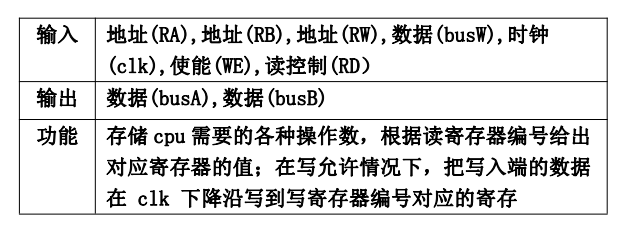
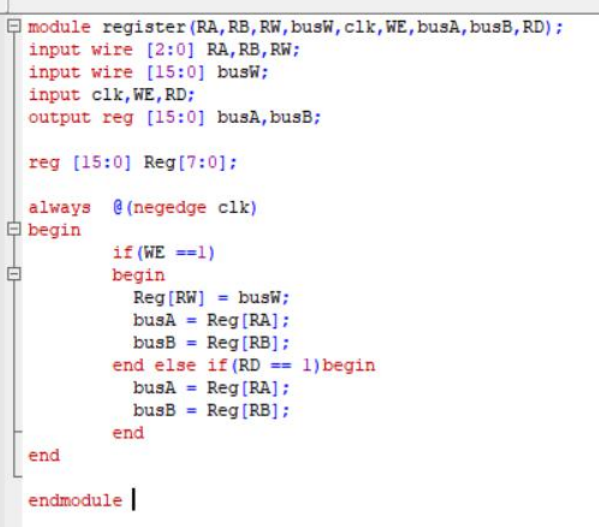


表 4.3 寄存器堆模块功能描述

Verilog 关键代码：



1. **ALU**

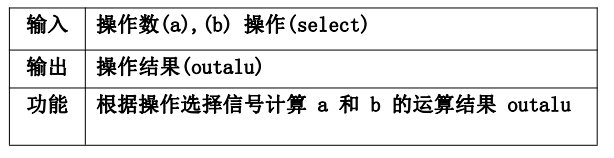
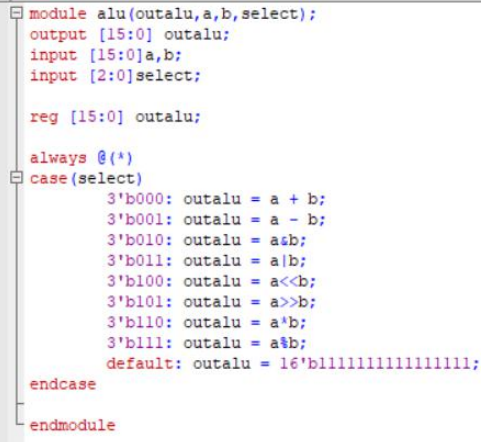


表 4.4 ALU 模块功能描述

Verilog 关键代码：



**5) 控制单元**

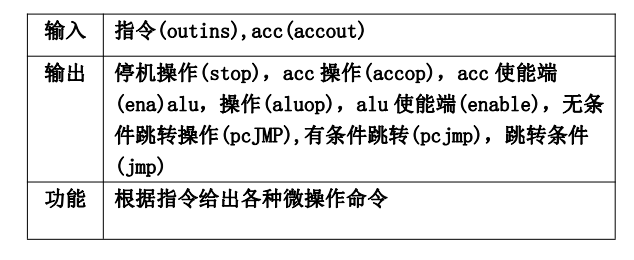
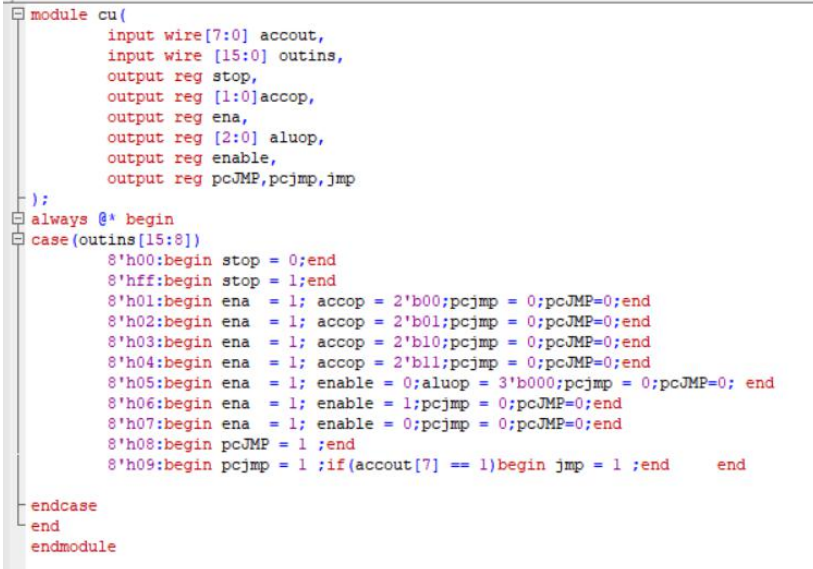


表 4.5 控制单元模块功能描述

Verilog 关键代码：



**4.2 CPU 顶层文件封装实现**

通过根据图 2 将以上定义的模块进行连接、封装就得到了目标 CPU，该 CPU 的输入为

系统时钟信号 clk 和重置信号 reset，输出 pc 是为了调试方便。

Verilog 关键代码：

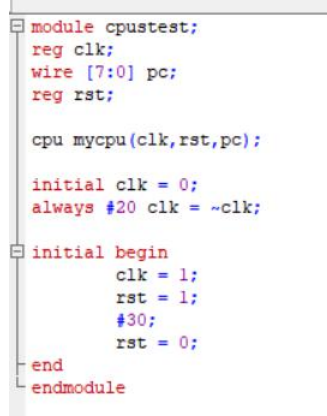


**4.3 CPU 模拟仿真**

为了仿真验证所实现的 CPU，需要定义测试文件并在测试文件中对指令存储器和寄存

器堆中的相应寄存器的值进行初始化，并通过仿真波形图查看是否指令得到了正确执行。

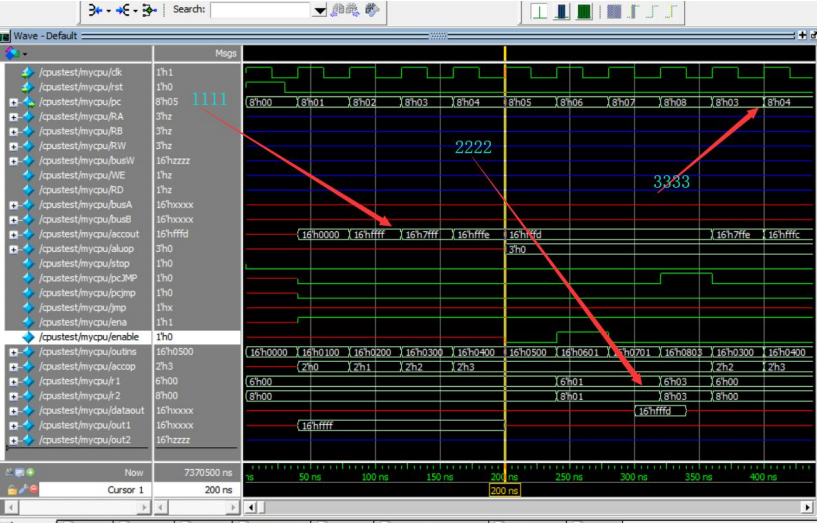
1. **TestBench 关键代码**



指令寄存器以 PC 给出的值为地址给出指令，因此再测试文件中只需要完成 PC 的控制就

可以了，clk 和 rst 定义为 1 后再延迟 30ns 是为了对 PC 进行初始化。

1. **ModelSim 仿真及分析**



由于十条测试指令都没有用到寄存器堆，所以均无数据

**最初**的 PC 给出的是 h00 是为了启动。停机指令与条件转移指令一起，在后文演示

 此处按 PC 给出数据依次演示如下:

**第一条指令**为清零指令，将 ACC 清零 **见 1111 处**

**第二条指令**为取反指令，将 ACC 取反 **见 1111 处**

**第三条指令**为算数右移指令，将 ACC 右移一位 **见 1111 处**

**第四条指令**为循环左移指令，将 ACC 左移一位 **见 1111 处**

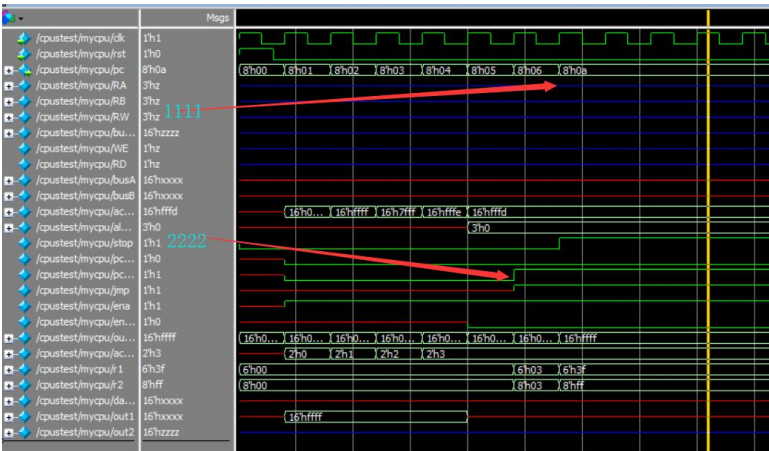
**第五条指令**为求和指令，和存储器 000000 位置数据进行求和 **见 1111 处**

**第六条指令**为置数指令，将 acc 存储到存储器 000001 位置 **见 2222 处**

**第七条指令**为取数指令，将存储器 000001 位置的数取出来，上升沿时候完成 **见 2222 处**

**第八条指令**为无条件跳转，跳转到 pc = 3 的位置，从 3 开始继续向后执行 **见 2222 处**

 条件跳转和停机指令演示：



第六条指令执行有条件跳转，acc 是 hfffd，最高位为 1，满足跳转条件(jmp==1)所以跳转(**见**

**2222 处**)，跳转到第 10 条停机指令，停机指令执行，pc 停止增长(**见 1111 处**)，各个数据也

停止变化。

**五、实验总结**

实验阶依次完成了四选一多路选择器，4 位全加器、8 位比较器、74138 译

码器，ALU，寄存器堆，PC，DataStorage，ins，最后的 ACC，CU 以及 CPU 的

封装和测试。

指令设计一共有 11 条，在实验要求外增加了开机 start 指令 h0000，指令中有很多位是没有使用的，可以留待以后拓展。