# 实验七 单周期CPU

## 实验目的、重点、难点

实验目的：

1.理解MIPS指令结构，理解MIPS指令集中常用指令的功能和编码，学会对这些指令进行归纳分类。

2.了解熟 MIPS体系的处理器结构，如哈佛结构的概念。

3.熟悉并掌握单周期CPU的原理和设计。

4.进一步加强运用verilog语言进行电路设计的能力。

5.为后续设计多周期cpu的实验打下基础。

实验重点：

熟悉使用verilog语言进行设计，仿真的过程。

实验难点：

观察仿真波形，确认电路设计是否正确。

## 实验任务

为了验证单周期CPU的正确性，我们需要建立一个SOPC(System on a programmable chip),在这个sopc系统中包含单周期CPU模块（single\_cycle\_cpu）和指令存储器（inst\_rom）,single\_cycle\_cpu从inst\_rom中读取指令，指令进入single\_cycle\_cpu开始执行，实现如框图2.1所示。single\_cycle\_cpu实现细节如图2.2所示。图2.2中除inst\_rom模块之外，其余的模块是single\_cycle\_cpu包含的模块。在图2.2中红色标明的信号线是single\_cycle\_cpu定义的外部端口信号。single\_cycle\_cpu的内部信号定义在图2.2中已经给出，内部信号用来连接译码模块，执行模块，regfile模块。图2.2中的执行模块和我们已经实现的ALU模块有区别，端口信号定义中，多了wd\_i,wreg\_i,wd\_o,wreg\_。

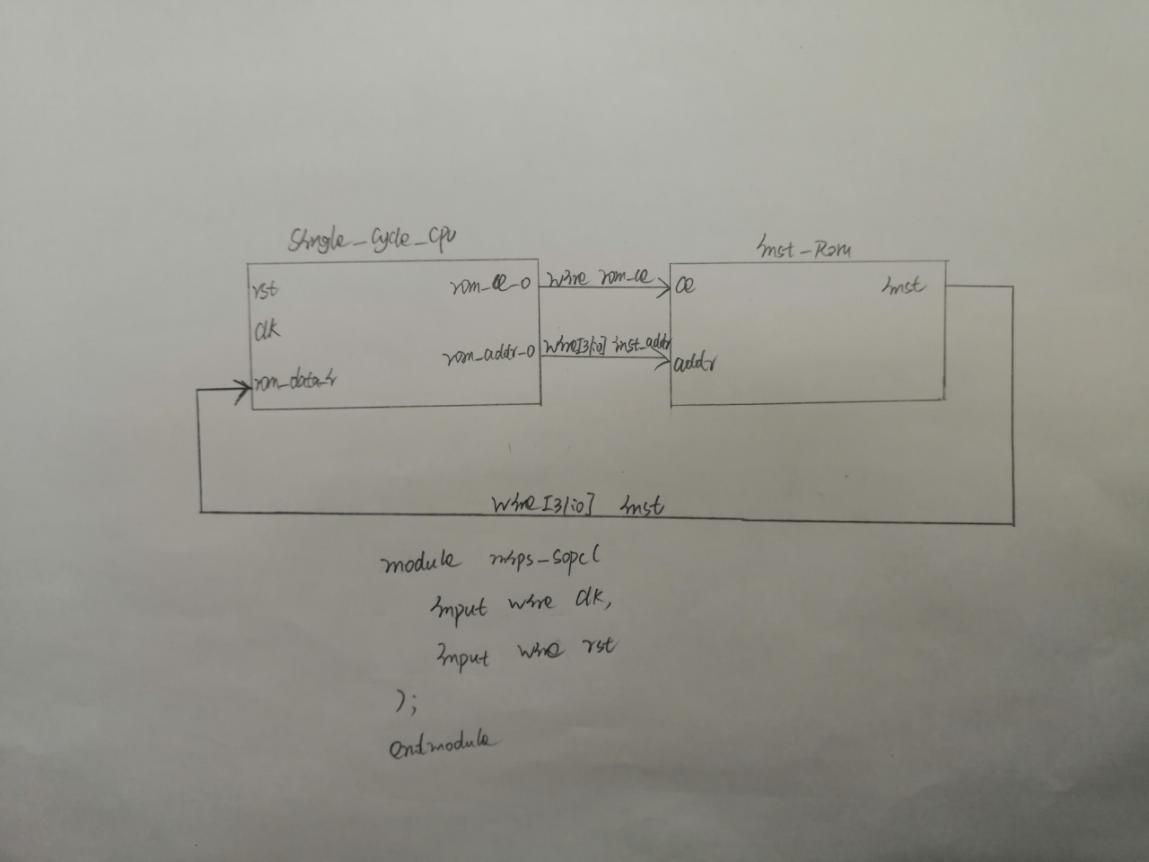


图2.1 sopc实现框图

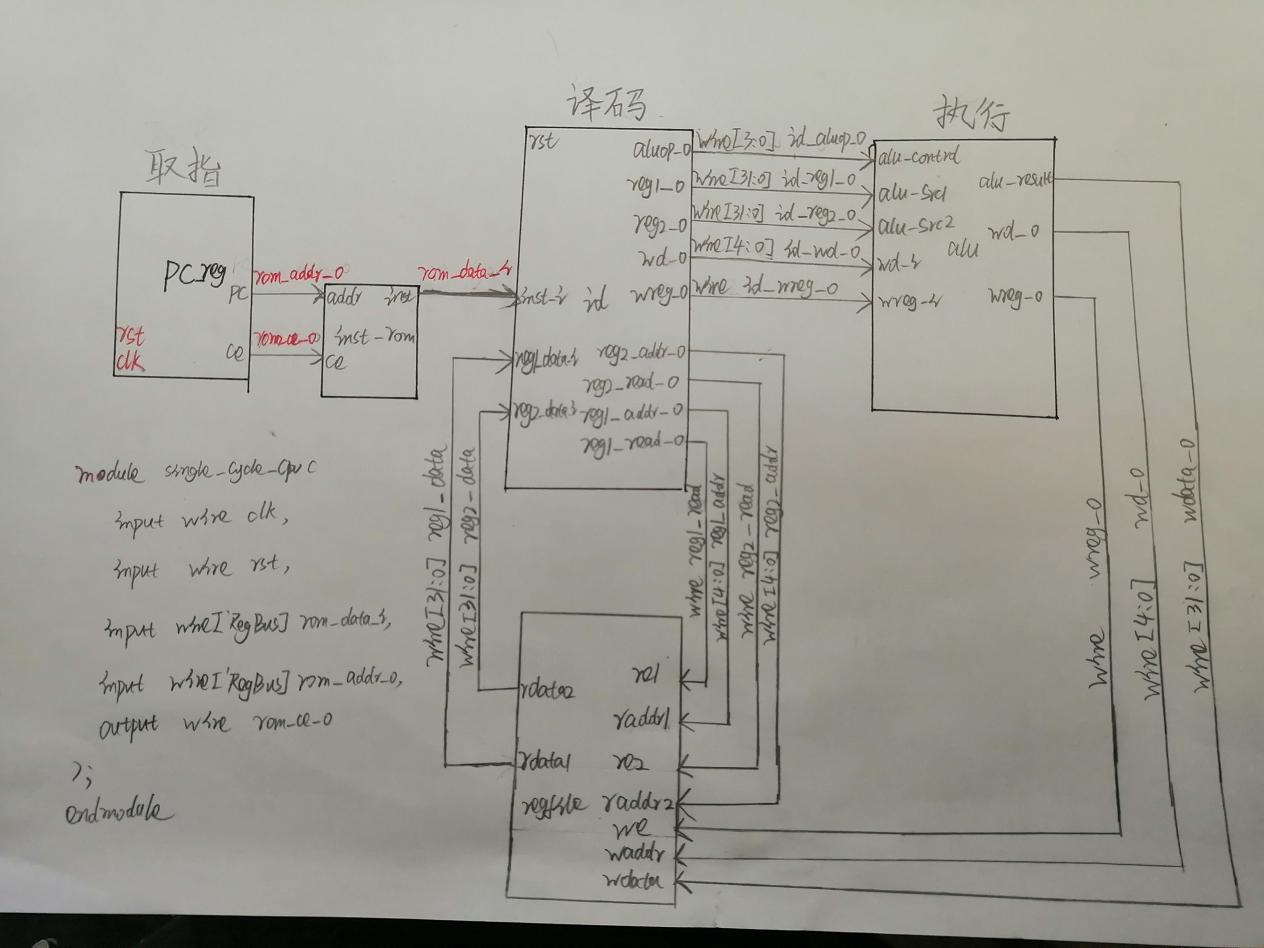
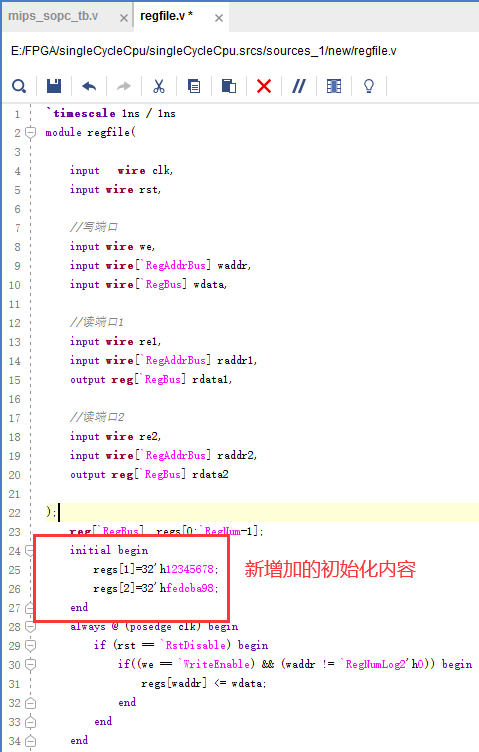
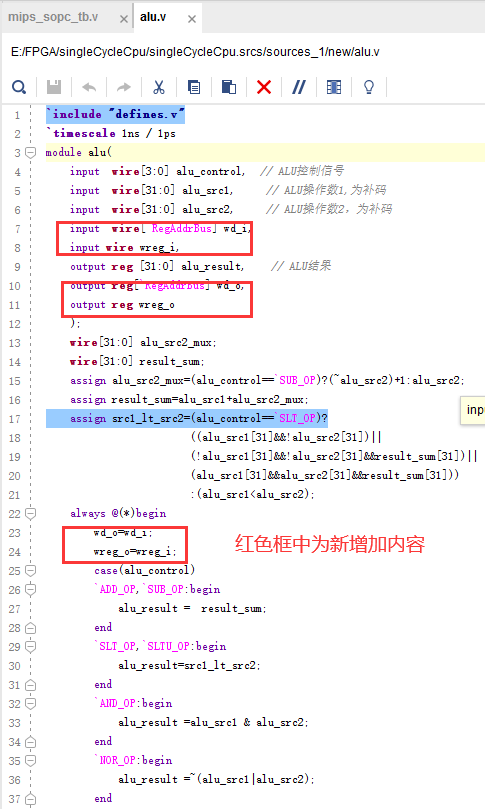


图2.2 single\_cycle\_cpu实现细节框图

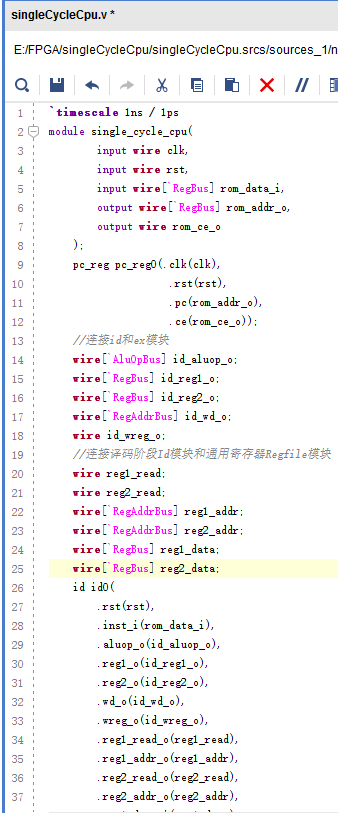
实现single\_cycle\_cpu，需要对之前实验课中已经完成的代码进行修改。regfile代码修改见代码2.1所示。alu代码修改见代码2.2所示。single\_cycle\_cpu代码见代码2.3和代码2.4所示。mips\_sopc代码见代码2.5所示。mips\_sopc\_tb代码见代码2.6所示。



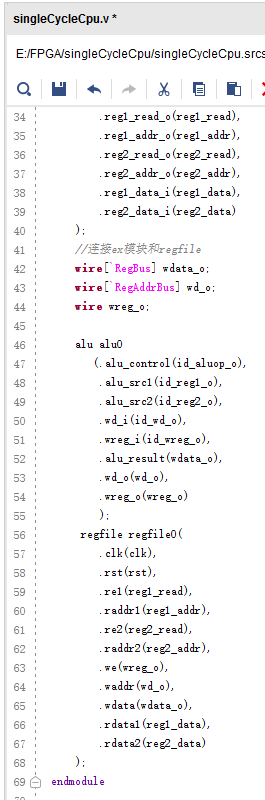
代码2.1 regfile代码修改



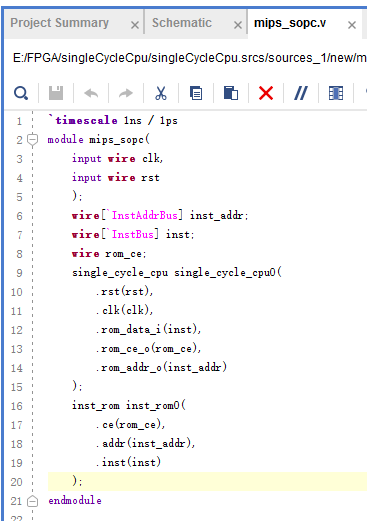
代码2.2 alu代码修改



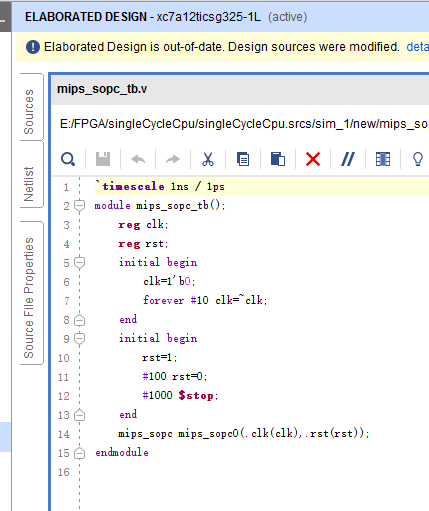
代码2.3 single\_cycle\_cpu代码1



代码2.4 single\_cycle\_cpu代码2



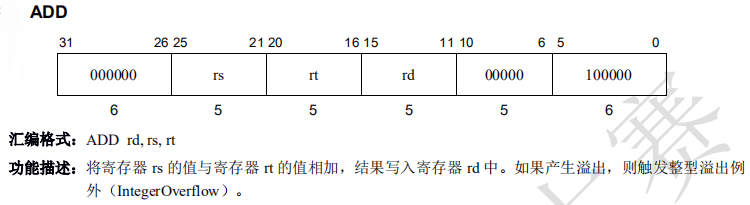
代码2.5 mips\_sopc代码



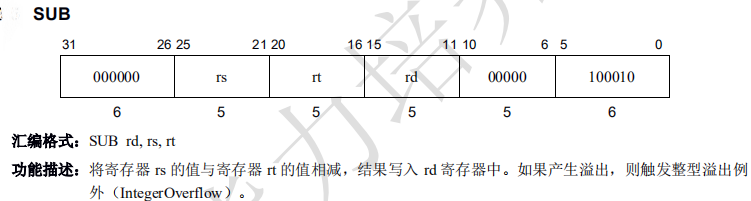
代码2.6 mips\_sopc\_tb代码

## 3、MIPS相关知识

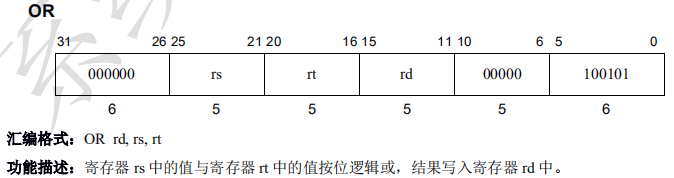
1. ADD指令。rd=rs+rt



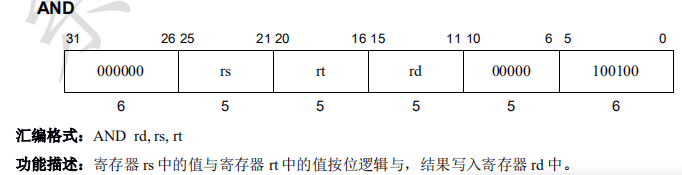
1. SUB指令。rd=rs-rt



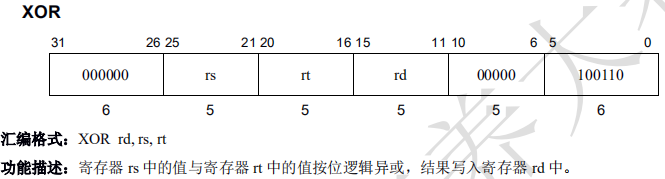
1. OR指令。rd=rs or rt



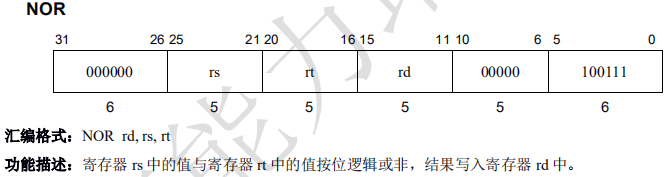
1. AND指令。rd=rs and rt



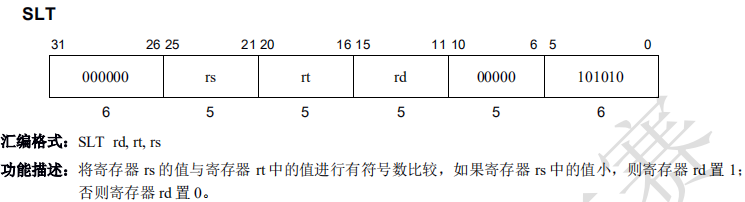
1. XOR指令。rd=rs xor rt



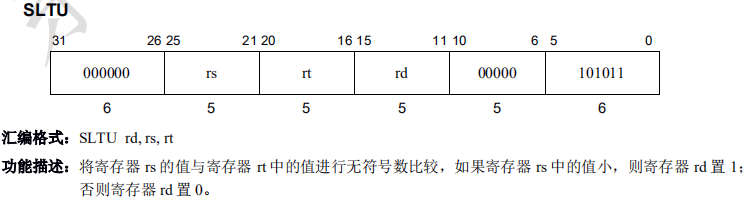
6、NOR指令。rd=rs nor rt



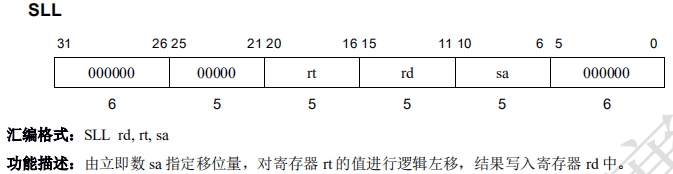
7、SLT指令。



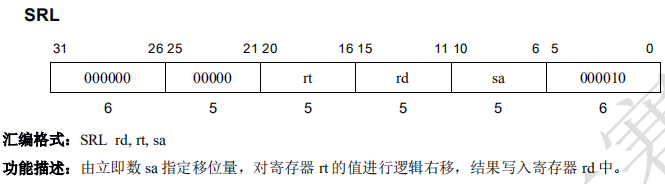
1. SLTU指令。



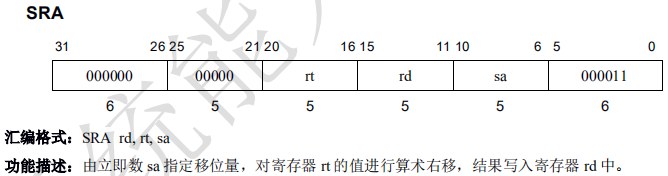
1. SLL指令



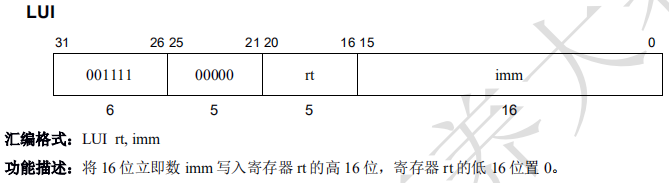
1. SRL指令



1. SRA指令



1. LUI指令



1. 指令格式归纳

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| op | rs | rt | rd | op2 | op3 |

31 26 25 21 20 16 15 11 10 6 5 0

