目录

[实验三 指令存储器rom上课讲解内容 2](#_Toc21236)

[1、实验目的、重点、难点 2](#_Toc19285)

[2、 实验任务实现 2](#_Toc17013)

[2.1 实现指令存储器 inst\_rom 2](#_Toc24187)

[2.2 取指模块实现 6](#_Toc24483)

[3、 mips相关知识 9](#_Toc4144)

# 实验三 指令存储器rom上课讲解内容

## 1、实验目的、重点、难点

实验目的：

1. 了解只读存储器ROM原理。

2. 理解ROM读取数据的过程。

3. 理解取指过程。

3.熟悉并运用verilog语言进行电路设计。

4.为后续设计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。观察rom的时序图。

## 实验任务实现

### 2.1 实现指令存储器 inst\_rom

inst\_rom设计图如图2.1所示，端口说明如表2.1所示。

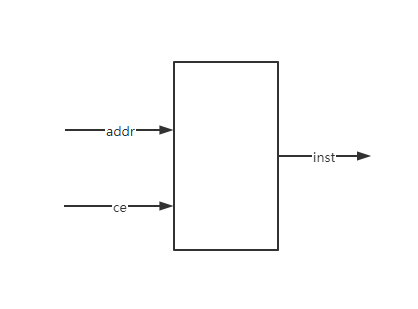


图2.1 inst\_rom设计框图

表2.1 inst\_rom 端口说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 接口名 | 宽度（bit） | 输入/输出 | 作用 |
| 1 | ce | 1 | 输入 | 使能信号 |
| 2 | addr | 32 | 输入 | 要读取的指令地址 |
| 3 | inst | 32 | 输出 | 读出的指令 |

指令的地址有程序计数器由PC给出，是按照字节进行寻址的，本次实验中设计的指令存储器是按照字进行寻址的，本次实验一共设计了128K\*4B指令存储器。

这里以4\*4B指令存储器举例，pc给出的地址范围为0-15，用4位二进制表示，而指令存储器的地址为0-3，用2位二进制表示，如图2.2所示。我们取pc地址时候去pc[3:2],0和1位舍弃。

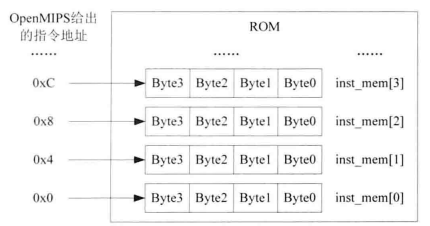
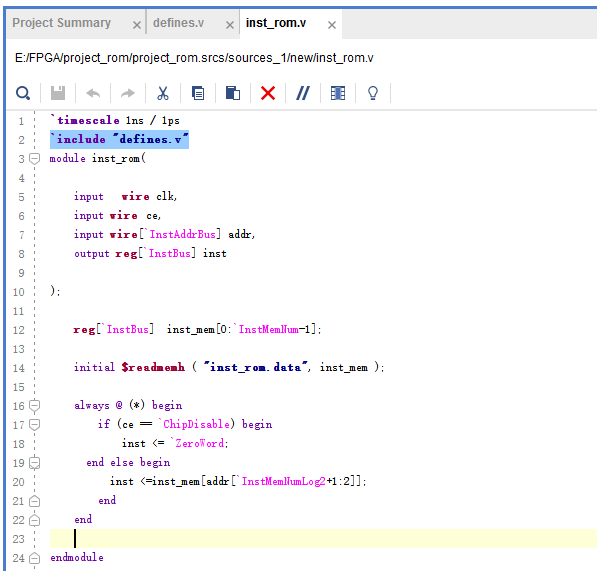
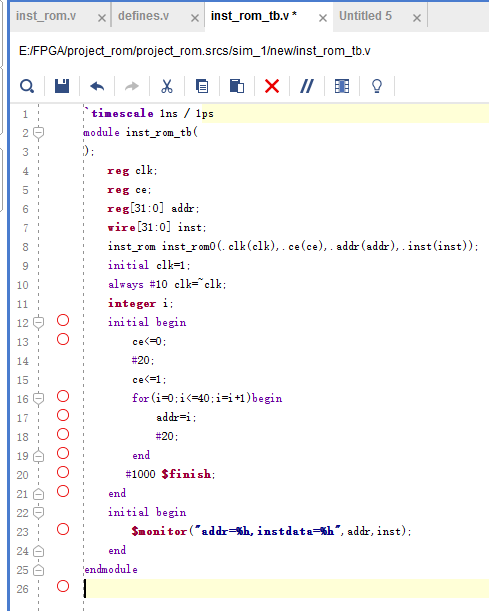


图2.2 pc 地址和rom地址关系

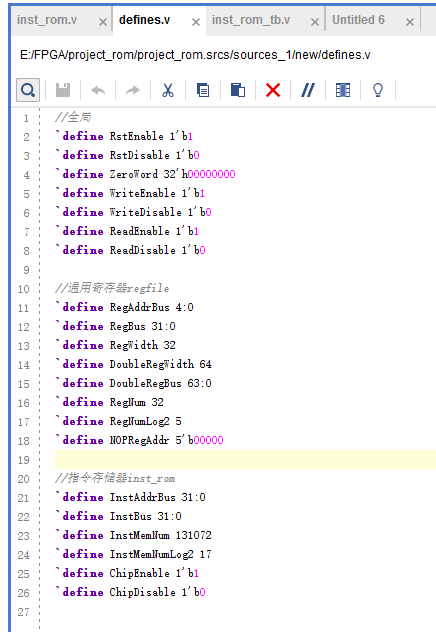
最终代码如代码2.1，2.2,2.3所示。



代码2.1 inst\_rom.v



代码2.2 inst\_rom测试代码



代码2.3 defines.v

### 2.2 取指模块实现

取指模块将pc模块和inst\_rom模块连成一个大的电路模块，实现框图如图2.3所示。自己写出端口描述。功能为每个时钟周期从指令存储器中取出一条指令。代码见代码2.4,2.5,2.6。

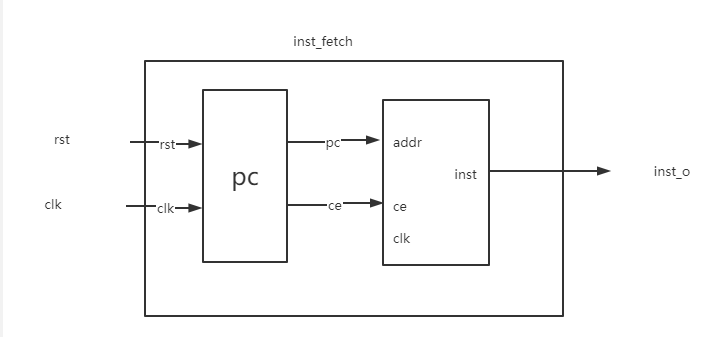
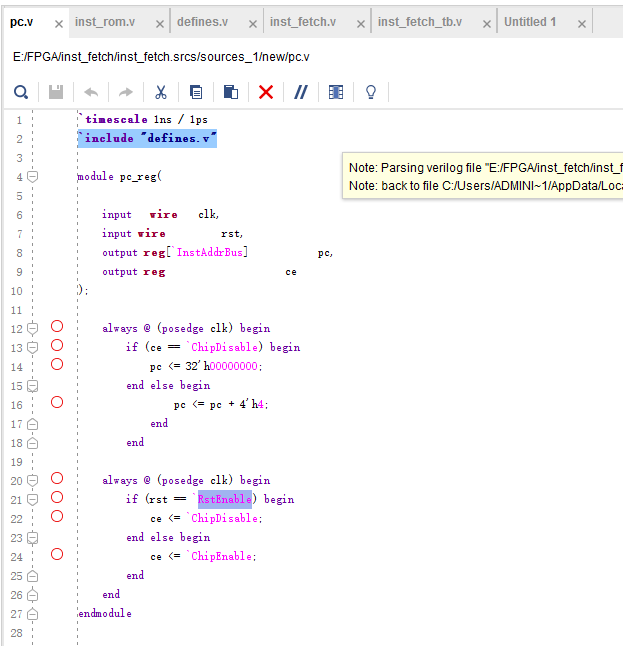
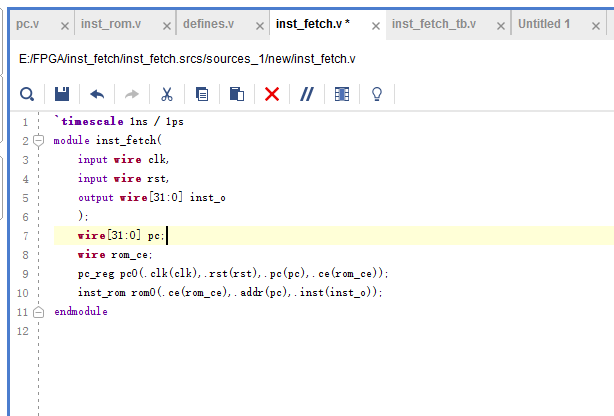


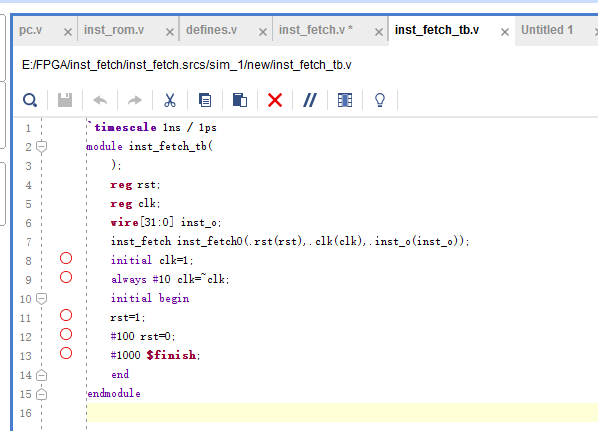
图2.3 inst\_fetch实现框图



代码2.4 pc设计代码



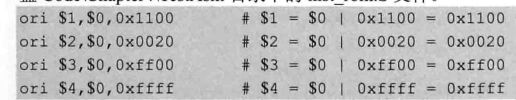
代码2.5 inst\_fetch设计代码



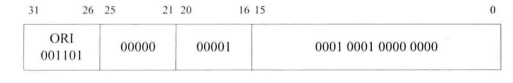
代码2.6 inst\_fetch设计代码

## mips相关知识

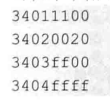
本次实验中用到的测试代码如下。



ori指令的格式如下所示。



转化出来的机器指令如下所示。



初始化rom用到的verilog指令为$readmemh。

$readmemh("<数据文件名（路径地址和文件名）>",<存储器名>);