实验1 简单计算机系统基本模块设计A-实验报告

电 25 吴晨聪 2022010311

# 1.1 ROM模块设计

**(1) 参照上面的步骤，实现对ROM模块【16位】的仿真验证；修改ROM中的初始化数据，给出仿真结果。**

16位ROM由cpurom 模块和addrGen 模块组成，两者的连接方式由下图1所示。其中cpurom 模块由rom.hex文件和Quartus自带的cpu 核组成，实现了输入一个地址，输出该地址寄存器内存储的数；addrGen 的具体实现见附件的addrGen.v 文件，输入为时钟信号、输出为地址，实现了在每一个时钟信号的上升沿地址均加一的功能。

一張含有 文字, 螢幕擷取畫面, 圖表, 行 的圖片

AI 產生的內容可能不正確。

图1 16位ROM模块的内部连接方式

测试代码为ROM\_tb.v，由于addrGen 输出的地址会随时钟信号自动变化，因此只需要加入变化的时钟信号，观察cpurom 输出结果即可，观察到的输出结果如图2所示，这与在rom.hex文件中事先存储好的内容是一致的，说明了16位ROM的可行性。

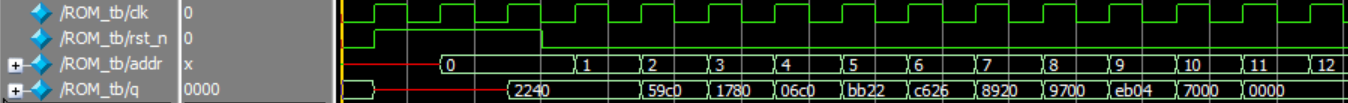


图2 16位ROM模块的测试结果

**(2) 建立32位位宽的ROM模块；给出仿真结果。**

与16位ROM模块相似，32位ROM由cpurom2模块和addrGen 模块组成，只需要把各模块的位宽改为32位即可。

一張含有 文字, 螢幕擷取畫面, 圖表, 行 的圖片

AI 產生的內容可能不正確。

图3 32位ROM模块的内部连接方式

测试代码为ROM2\_tb.v，观察cpurom2 输出结果如图4所示，这与在rom2.hex文件中事先存储好的内容是一致的，说明了32位ROM的可行性。



图4 32位ROM模块的测试结果

# 1.2 寄存器设计

**(1) 参照上面的步骤，实现对寄存器组模块【8位，4个】的仿真验证；分析仿真结果。**

寄存器模块由regfile.v实现，实现了在寄存器内部的读写功能。有两个读通道，输入寄存器内部地址可以输出该地址存储的数；还有一个写通道，当reg\_we=1时表示写通道打开，此时会将di存储到寄存器的nd地址内，由此实现读和写的功能。

寄存器模块的测试代码为regfile\_tb.v(regfile2\_tb.v)，该测试代码首先向寄存器的地址0、1、2、3中分别存入10、11、12、13，然后通过读通道依次读出这四个通道的数字，发现等于写入的数字，表示寄存器运行正常。

一張含有 螢幕擷取畫面, 多媒體軟體, 軟體, 文字 的圖片

AI 產生的內容可能不正確。

图5 8位4个寄存器模块的测试结果

**(2) 建立32位位宽，包含32个寄存器的寄存器组模块；给出仿真结果。**

与8位4个寄存器相似，寄存器模块由regfile2.v实现，寄存器模块的测试代码为regfile2\_tb.v，通过读通道依次读出这四个通道的数字，发现等于写入的数字，表示寄存器运行正常。

一張含有 螢幕擷取畫面, 行 的圖片

AI 產生的內容可能不正確。

图6 32位32个寄存器模块的测试结果

# 1.3 ALU设计

**(1) 参照上面的步骤，实现对alu模块【8位】的仿真验证，分析仿真结果。**

8位alu对应的模块为alu.v，主要实现了计算机内部的基本计算功能。输入两个运算数、运算符、进/借位即可输出运算结果和进/借位，此外还会输出一个零标识符来表示运算结果是否为0。实现的方法就是先对输入的符号(cs)进行判断，得出要进行的是何种运算方式，再根据不同的运算方式分别执行该运算方式对应的代码即可。

alu\_tb.v为8位ALU模块的测试代码，通过设置不同的算例来将7种运算方法遍历，来检验该运算器的各个功能是否能可靠运行。根据测试代码的运行结果，发现运算结果正确，这说明了该alu能正常运行。

一張含有 螢幕擷取畫面, 多媒體軟體, 軟體, 文字 的圖片

AI 產生的內容可能不正確。

图7 8位ALU模块测试结果

**(2) 建立32位alu模块，给出仿真结果。**

32位alu对应的模块为alu2.v，alu2\_tb.v为32位alu模块的测试代码。根据测试代码的运行结果，发现运算结果正确，这说明了该ALU能正常运行。

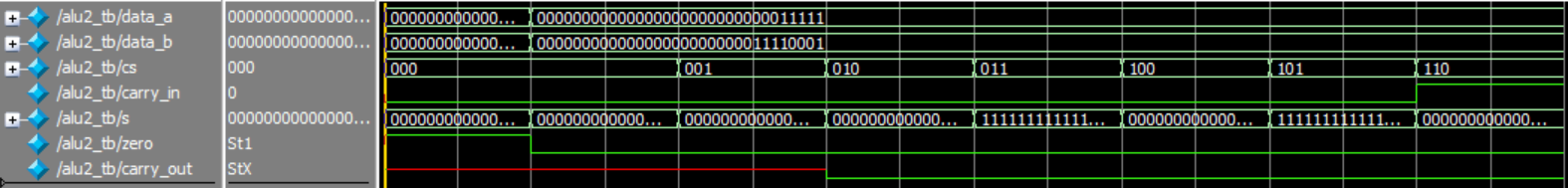


图8 32位ALU模块测试结果