

实验四——PC寄存器与数据存储器

实验内容

使用Verilog设计实现PC寄存器和数据存储器

文件内容

- ./实验报告.pdf: 该实验的实验报告。
- ./code文件夹: 储存实验相关代码的文件夹。
 - ./code/ProgramCounter.v: PC寄存器的Verilog代码。
 - ./code/ProgramCounter_tb.v: PC寄存器的测试代码。
 - ./code/DataMemory.v: 数据存储器的Verilog代码。
 - ./code/DataMemory_tb.v: 数据存储器的测试代码。

实验步骤

PC寄存器

- 程序实现

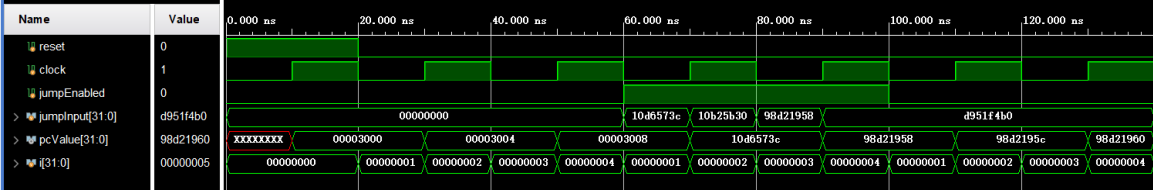
使用always实现时序逻辑，对时钟周期的上升沿敏感。

先判断reset标志。如果该信号被设置为1，则恢复所有设置，将pcValue的值设置为起始地址0x00003000

然后判断jumpEnabled标志。如果该信号被设置为1，表示此时需要进行跳转操作，将pcValue的值设置为jumpInput的值，实现跳转。

最后，如果上述两个标志都是0，则进行正常跳转，将pcValue的值加4。

- 测试结果



每个时钟周期上升沿，如果jumpEnabled没有被设置为1，PC寄存器都能自己跳转到下一条指令。在设置了jumpEnabled为1之后，PC寄存器会跳转到jumpInput指向的地址。因此，该PC寄存器是正确的。

数据存储器

- 程序实现

使用always实现时序逻辑，对时钟周期的上升沿敏感。

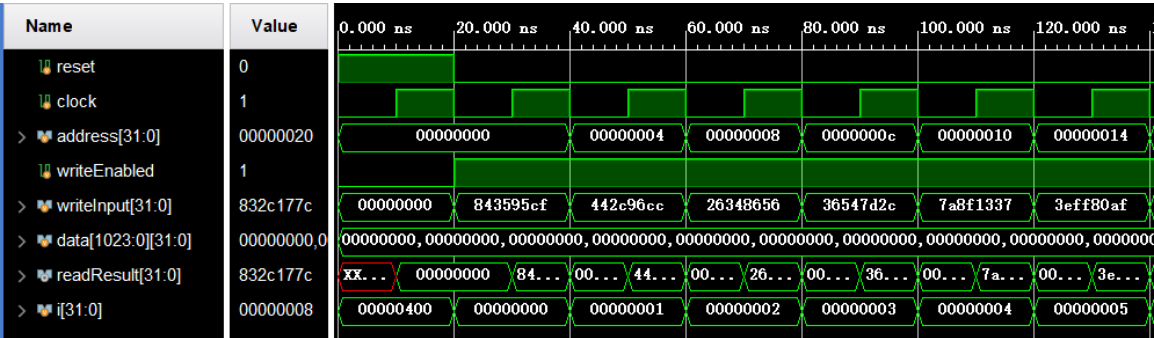
先判断reset标志。如果该信号被设置为1，则恢复所有设置，即将data中的所有位置清零。

然后判断writeEnabled标志。如果该信号被设置为1，则进行写入，将writeInput中的数据写入到address[31:2]的位置中。

由于readResult为组合逻辑，所以其不再always模块内实现，在外面直接赋值即可。当reset被设置或writeEnbaled被设置时，readResult的结果无意义。

- 测试结果

写入测试：写入测试在仿真刚开始时进行，对1024个位置依次进行写入，此处截取前5个位置的仿真图像。由于该程序为时序逻辑，设置了始终上升沿敏感。在clock的上升沿处，电路才进行工作，正确返回输入的数值。因此写入是正确的。



读取测试：读取测试是紧跟在写入测试后进行的，依次对1024个位置进行读取测试，此处截取前5个位置的仿真图像。可以对比之前的写入测试，读取时可以正确获得对应位置的值。同时，根据时钟沿可以判断，读取为组合逻辑。因此读取测试时正确的。

