**多周期MIPS处理器设计**

实验要求：

参考提供的单周期MIPS处理器，完成支持7条指令的MIPS处理器。

实验成果：

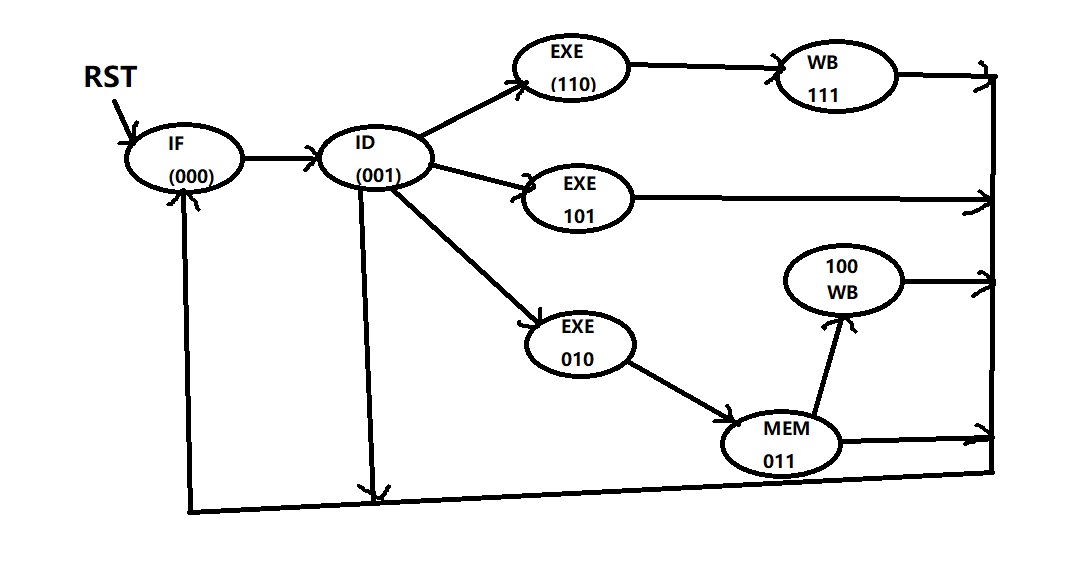
见Main-verilog文件，其中Main.v为顶层文件，cpu\_tb.v为测试文件。

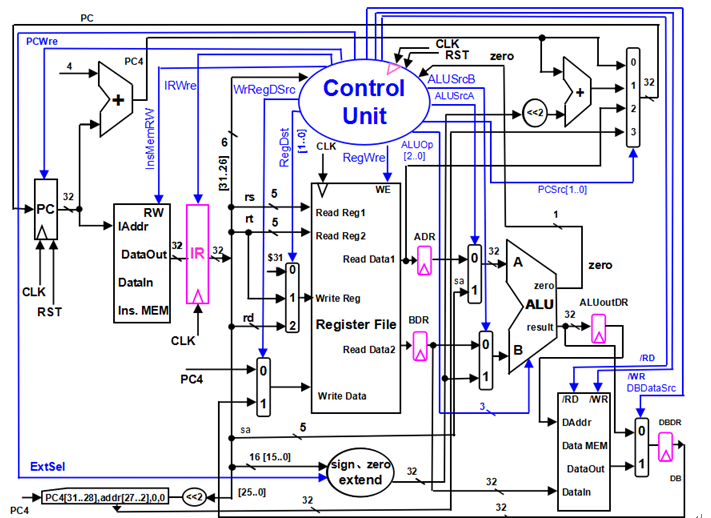
设计思路：

使用verilog语言，实现add sub or sw lw beq j七条指令。多周期CPU指的是将整个CPU的执行过程分成几个阶段，每个阶段用一个时钟去完成，然后开始下一条指令的执行，而每种指令执行时所用的时钟数不尽相同，这就是所谓的多周期CPU。CPU在处理指令时，一般需要经过IF（取指令） ID（指令译码） EXE（指令执行） MEM（存储器访问） WB（结果写回）五个阶段。一条指令的执行最长需要五个时钟周期。

指令处理过程：IF—ID—EXE—MEM—WB—IF

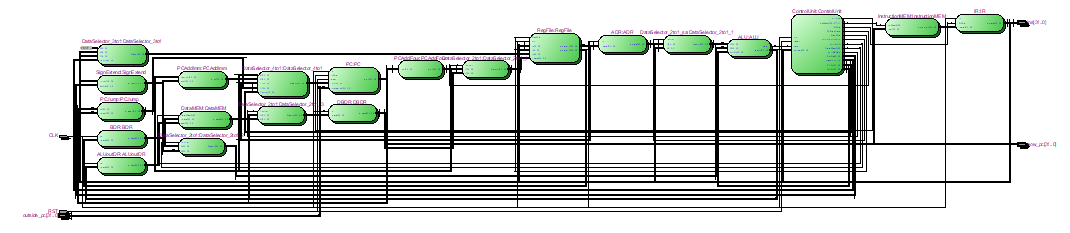
下图为多周期CPU状态转移图：



设计三个D触发器用于保存当前状态，是时序逻辑电路，RST用于初始化状态“000”，两个组合逻辑电路一个用于产生下一阶段的状态，另一个用于产生每个阶段的控制信号。下一状态取决于指令操作码和当前状态；而每个阶段的控制信号取决于指令操作码、当前状态和反映运算结果的状态zero标志。

上图为数据通路及控制线路图。（转自CSDN）

由QuartusII生成的RTL图为：



cpu\_tb.v仿真结果：

