Introduction to Digital Logic Design Lab

EECS 31L

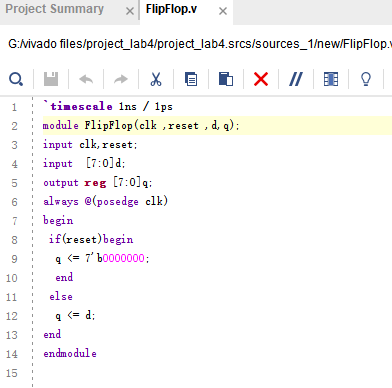
Lab 4

**1,** Objective

In this lab, I designed 3 models: D flip flops, Instruction Memory, and Register File, which all are the fundamental components of a Single-cycle Processor.

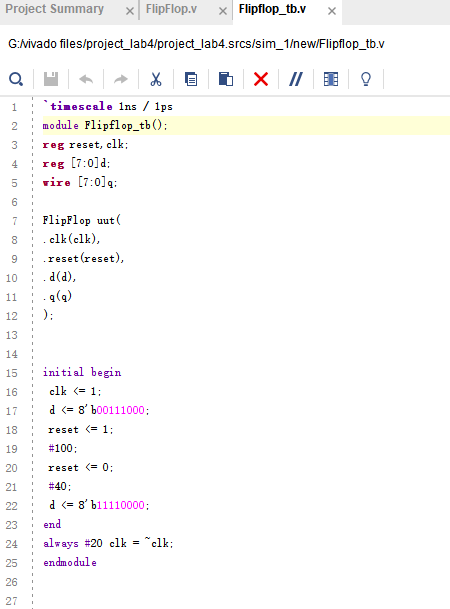
**2,** Procedure and result

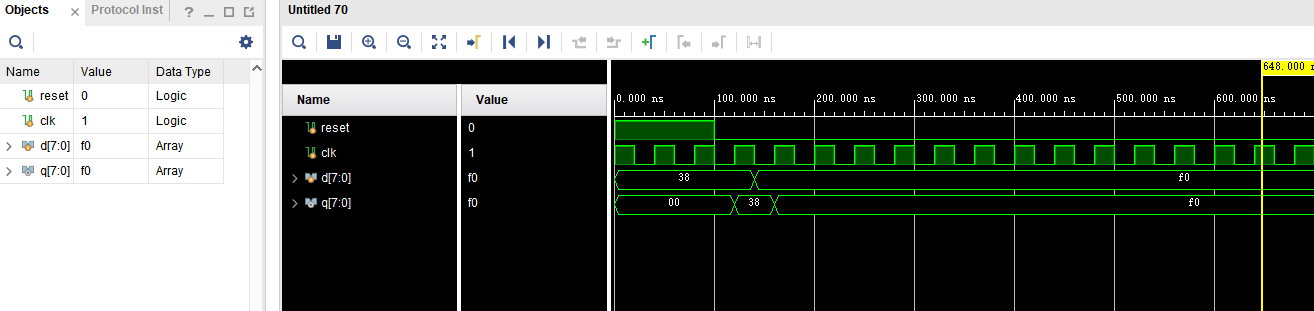
**A: D flip flop:**



D flip flop’s output change when the clock rises and become zero when reset is activated, so the design is quite straight forward.

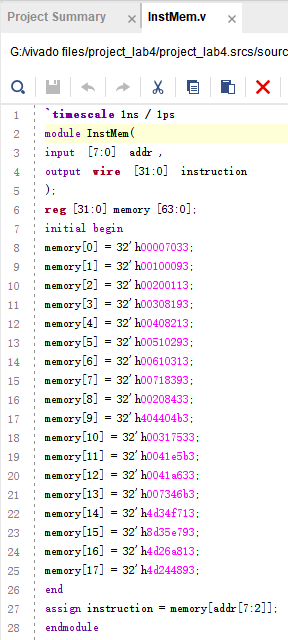
**Testbench and result for D flip flop**





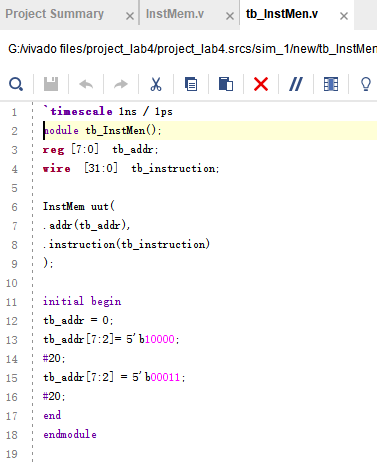
**B,** Instruction Memory

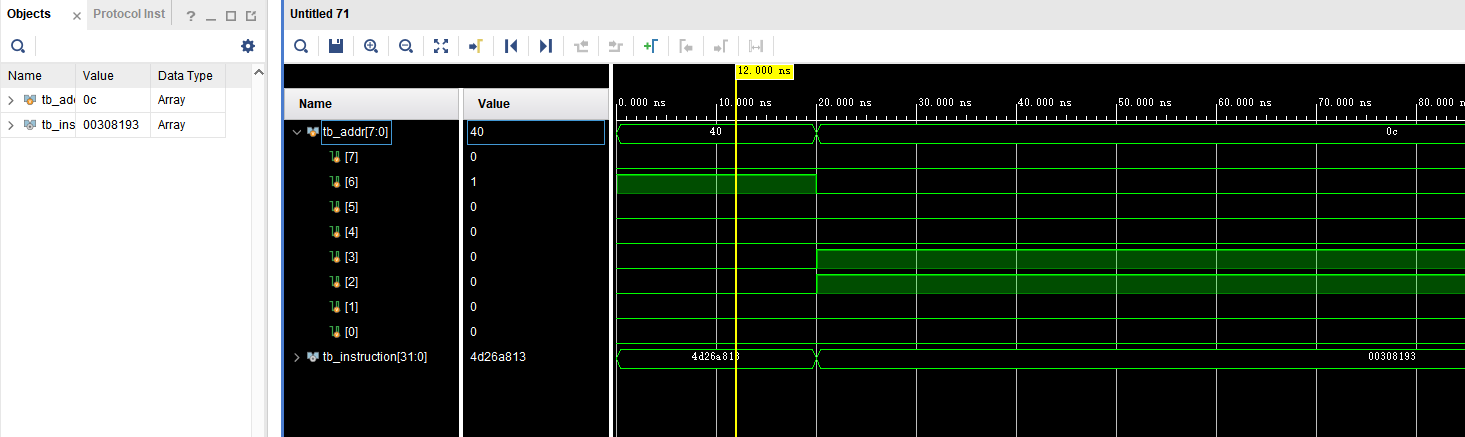
Instruction memory is a component that extract instructions from the memory based on the input address. Therefore, firstly I need to define a memory to store instructions, which is called Memory, and them I need to assign the selected memory row’s content to output.



**Testbench and result for InsMen**

As expected, the output shows the correct instruction stored in the according memory.

****

****

**C,** Register file

Register file is a component that is able to read content from registers or write content to registers. Although I do get the correct output, I am still confused. Because the register memory is either a input or output, it won’t appear on waveform as other inputs and outputs, and when I tried to connect the memory I defined in the testbench, it said “cannot access memory directly”, I tried for a long time and still couldn’t let the memory show in the waveform as shown in LAB4. Eventually, what I did is directly write almost the whole program in testbench.

