# Digital Logic Design

# Computer Assignment 5

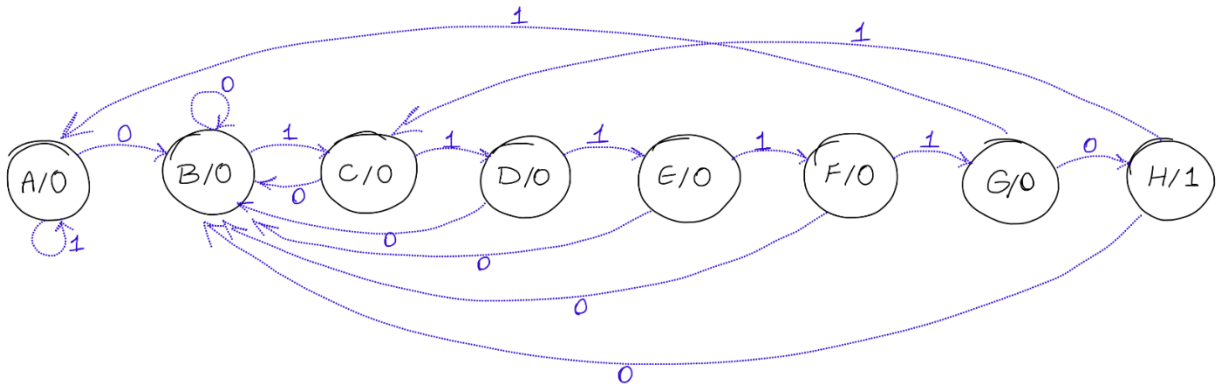
# Report sheet

Niloufar mortazavi

220701096

نیلوفر مرتضوی

First we design a sequence detector using moore machine like below:



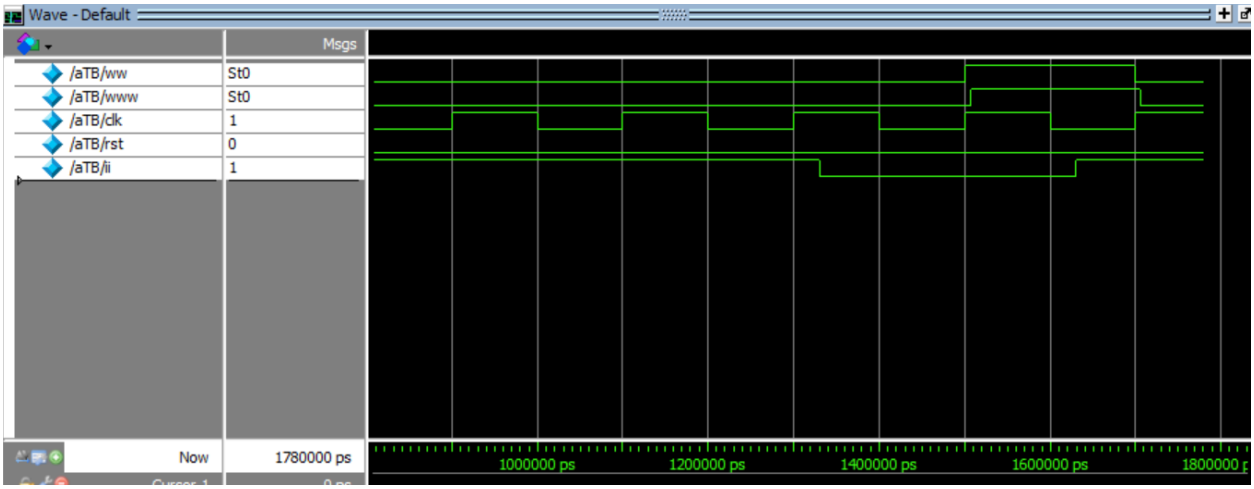

## state table

| state | 0 | 1 | w |
|---|---|---|---|
| A | B | A | 0 |
| B | B | C | 0 |
| C | B | D | 0 |
| D | B | E | 0 |
| E | B | F | 0 |
| F | B | G | 0 |
| G | H | A | 0 |
| H | B | C | 1 |
| | $state^+$ | | |

## transition table

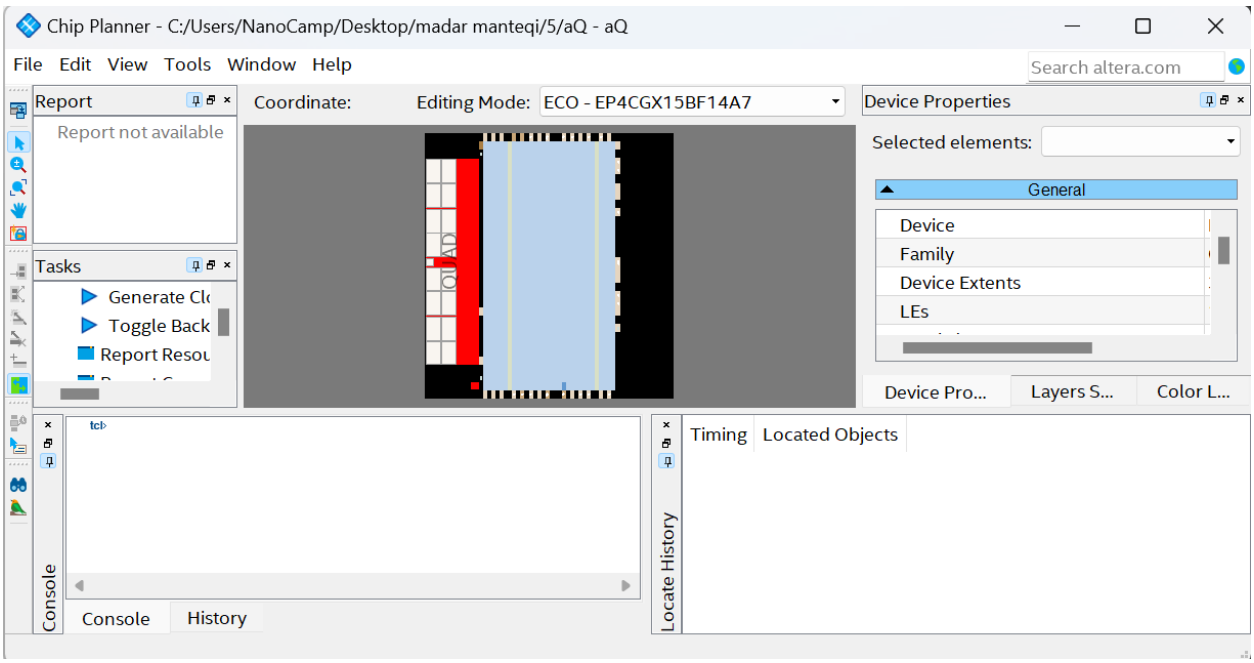
| $V_2V_1V_0$ | 0 | 1 | w |
|---|---|---|---|
| 000 | 001 | 000 | 0 |
| 001 | 001 | 010 | 0 |
| 010 | 001 | 011 | 0 |
| 011 | 001 | 100 | 0 |
| 100 | 001 | 101 | 0 |
| 101 | 001 | 110 | 0 |
| 110 | 111 | 000 | 0 |
| 111 | 001 | 010 | 1 |
| | $V_2^+V_1^+V_0^+$ | | |

We write the Verilog using the designed states. The we wrote a testbench and we synthesis the Verilog code using quartus and by writing a testbench to compare pre and post synthesized code we can see the waveform below:

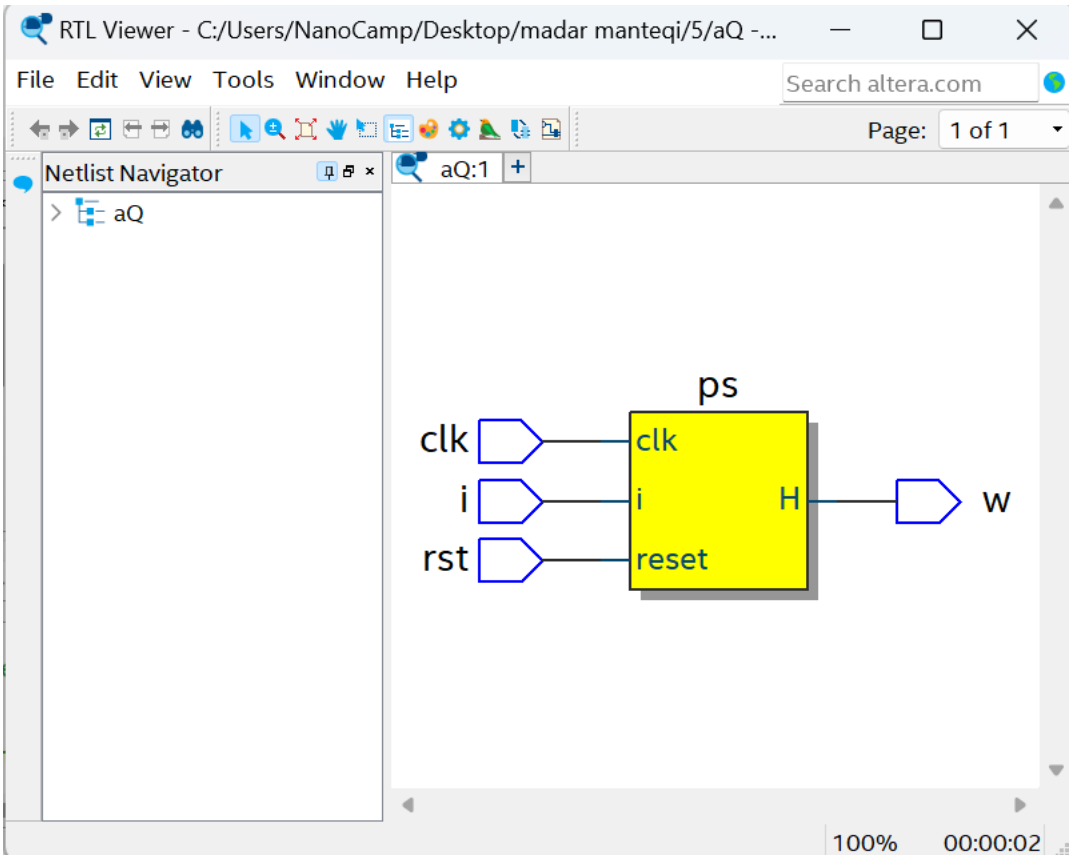


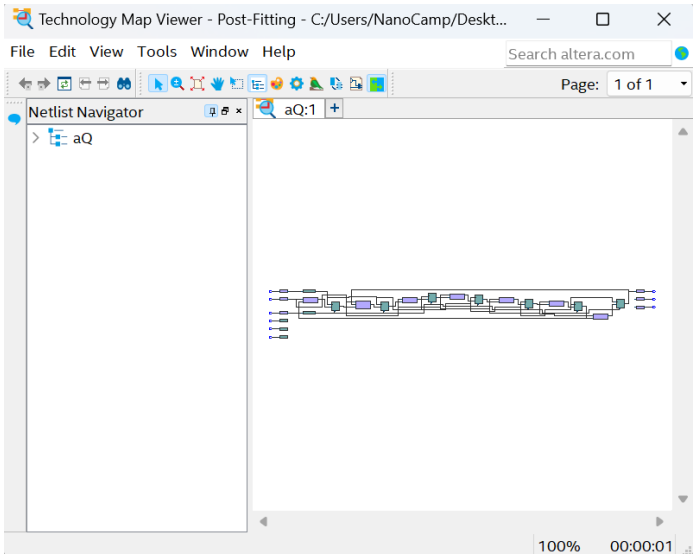

as we can see you can detect some timing differences because we have delays in the synthesized code.

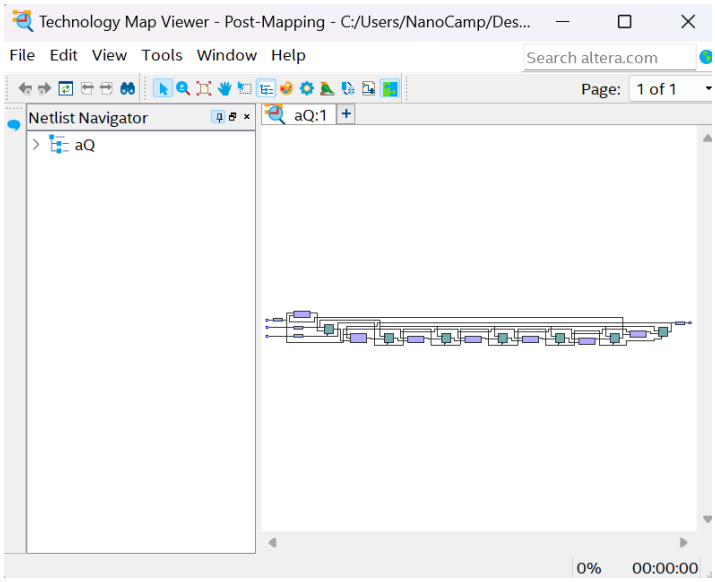
We can acces the floor plan:

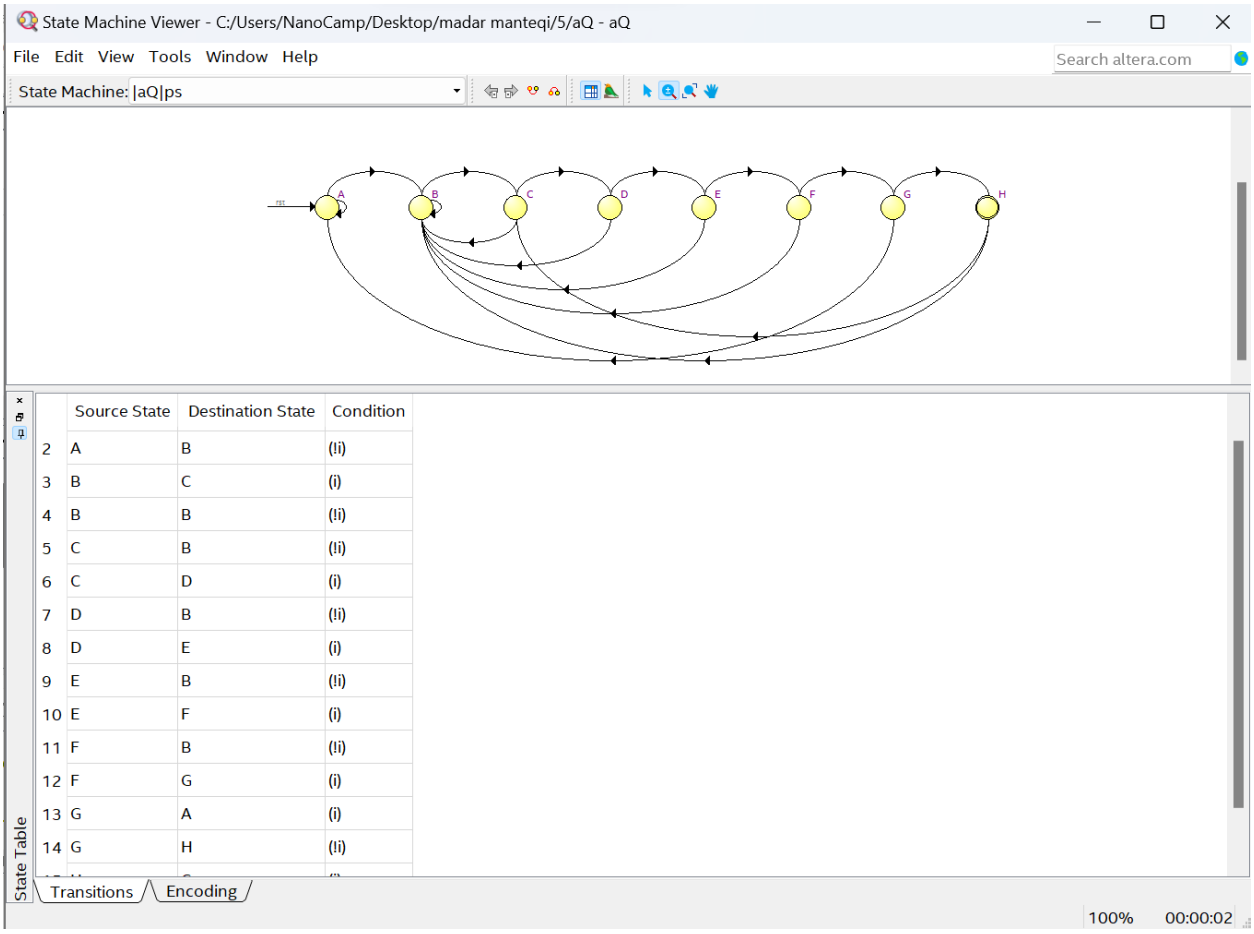


We can see the netlists in netlist viewer:

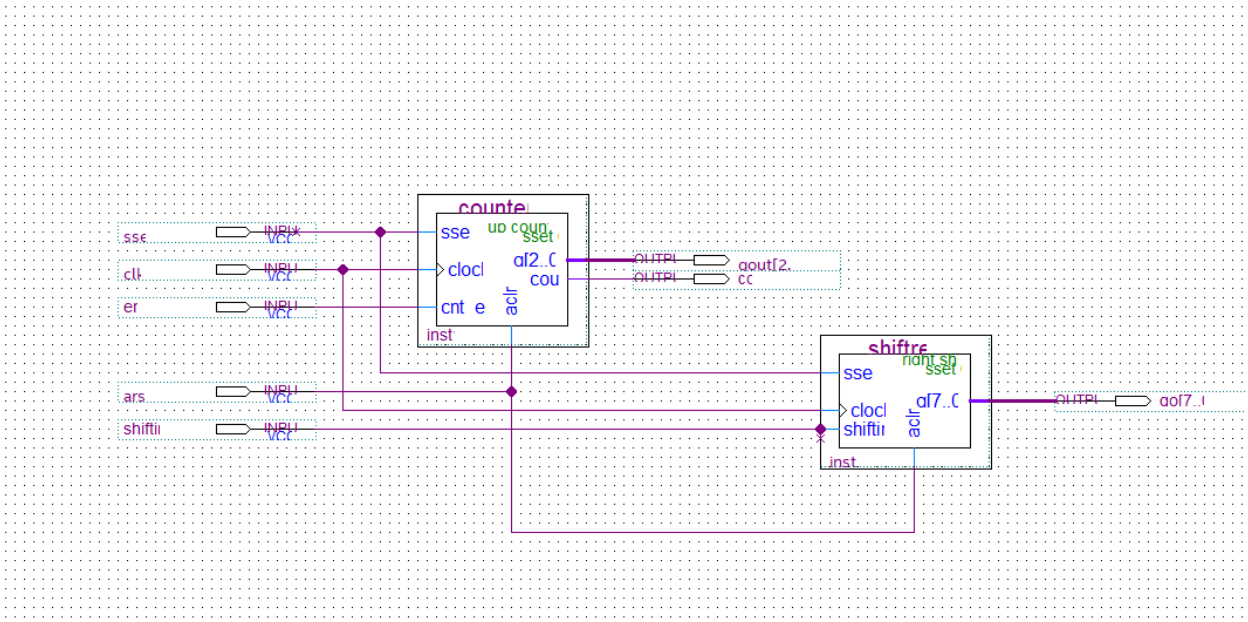


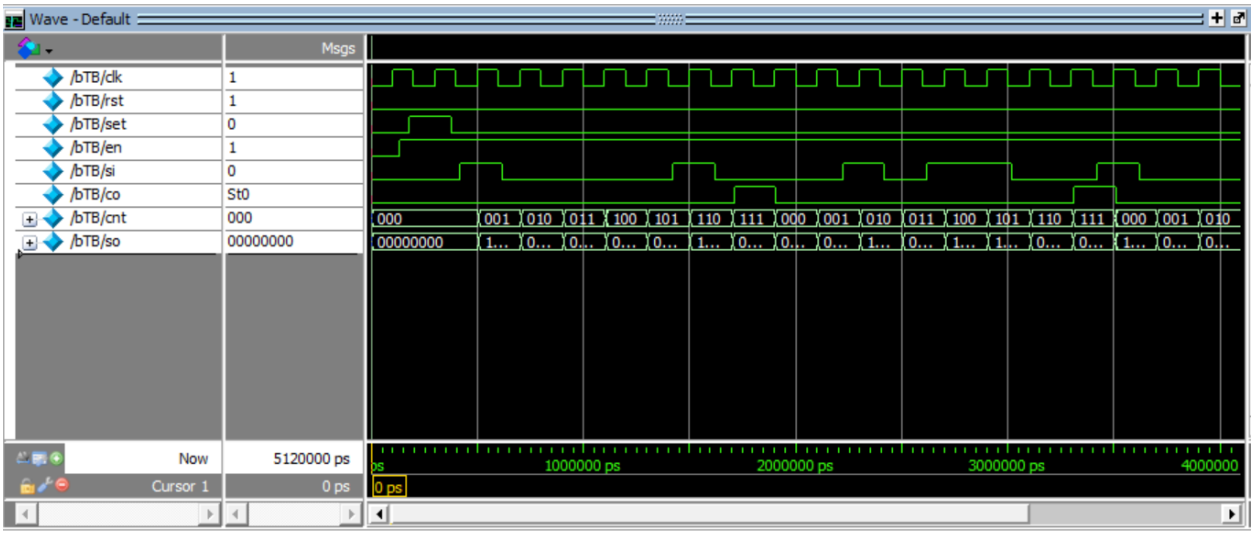






For part b we use the Quartus library and by changing the register and the counter we built a reg and bit counter.






same as the last part we observe the plan floor and the netlist.