Brunel University London

重庆邮电大学

CHONGQING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

TERM: Spring,2022

Module: EE1616 Electronics Workshop

CLASS: 34092102

BRUNEL ID: 2161047

CQUPT ID: 2021215069

NAME: Xukang Liu

TUTOR: Zhipeng Wang

April 17, 2022

Introduction and aims:

This seminar we need learned something about sequential logic device. I learned the principle of J-K flip-flop, and understand the function of 5 pins in J-K flip-flop. Using JKFF as a symbol created more flip-flops which derived from the JKFF. Furthermore, using JKFF and its derived flip-flops designed synchronous and asynchronous counters and test it.

Task description:

1. Create a new project add a JKFF symbol. Add 5 inputs and 1 output. Connect them by a correct way. Then create a VWF to simulate it, note that PRN needs to be high level all the time and CLRN should be set to 0 for a short time at the begging.

2. **Asynchronous trigger flip-flop.** In the waveform, set J and K to 1 all the time and simulate. Observe the result, Q need to be a periodic wave and the period needs to be twice the clock signal.

3. **Synchronous trigger flip-flop**. Connect the J and K together as an input. Create a waveform and compare it to the asynchronous trigger flip-flop.

4. **Delay flip-flop**. Based on the synchronous trigger flip-flop, place a not gate at the input, so the delay flip-flop can connect input D to J and NOT D to J.

5. **4-bit binary counter.** When it comes to counters,there is a law that when add 1 bit the corresponding counter will add a JKFF. So 4-bit binary counter need 4 JKFF. Connect J and K together, and J and K inputs of each JKFF are all determined by the initial input. Each output signal Q is used as the clock signal of the next JKFF. This 4-bit counter is asynchronous counter.

6. **Synchronous TFF binary counter.** Unlike asynchronous binary counters, this counter require all the clocks together, it means all JKFF will work at the same time. The output of the previous JKFF is used as the input of the next JKFF.

Experimental method:

1. Create a new project and add JKFF in it.

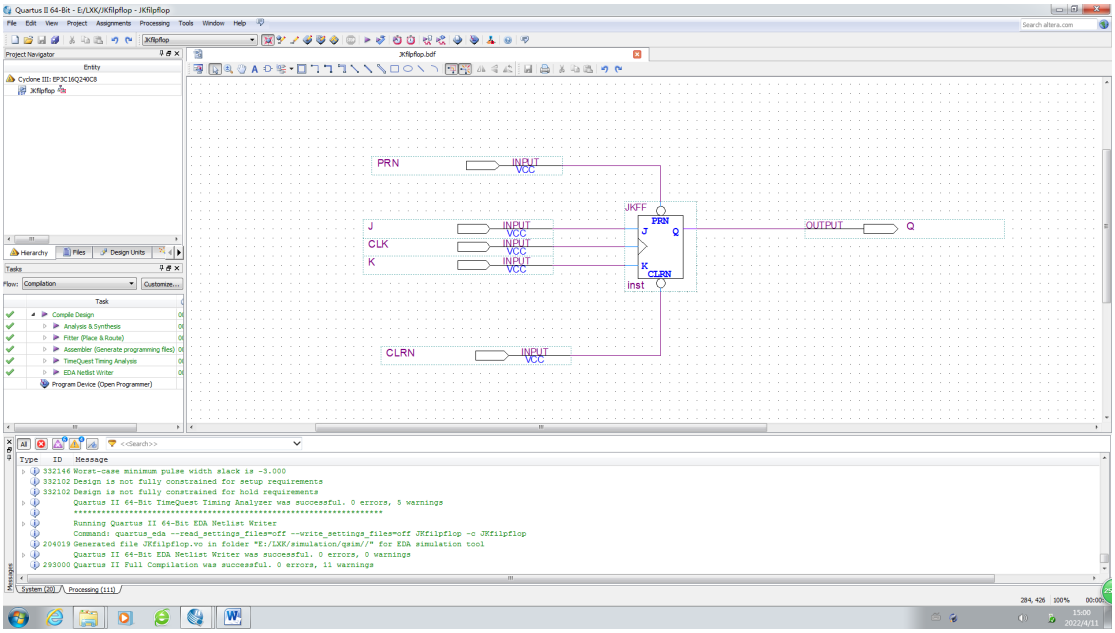
2. Draw the circuit diagram of each part according to the given requirements.

3. Compile and simulate it, then we get waveform of each part.

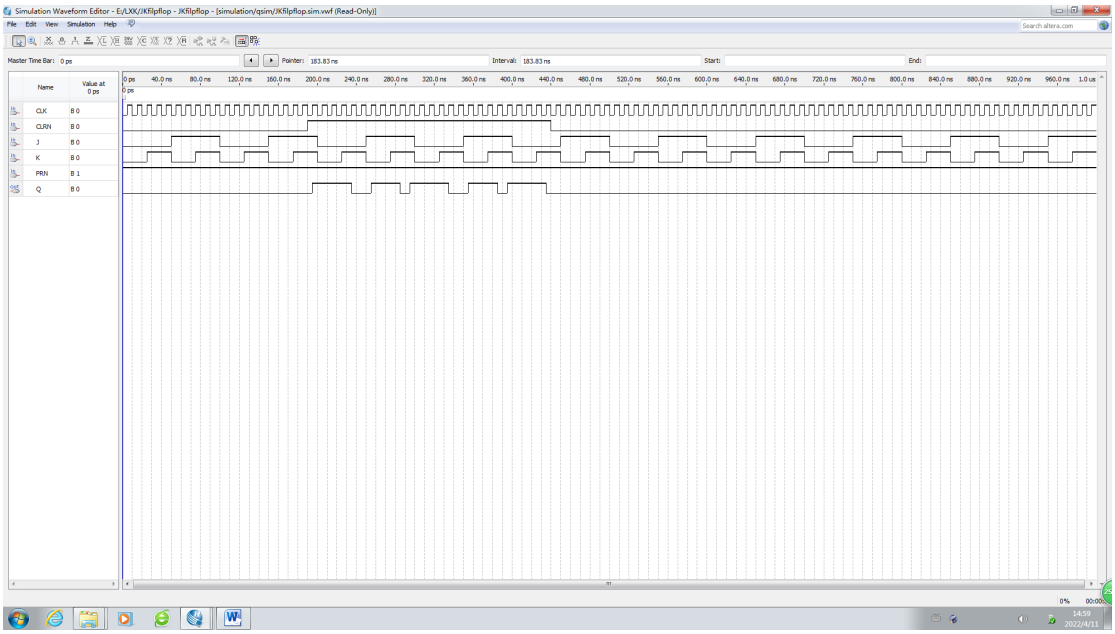
4. Check whether the waveform diagram is correct according to the truth table.

Results and observations:

PART1 create a JKFF and simulate it.



This is the waveform



Check with the truth table:

Point 1: at 235.0ns

CLK 0⟶1 J=0 K=1 Q 1⟶0 Correct

Point 2: at 395.0ns

CLK 0⟶1 J=1 K=1 Q 0⟶1 Correct

……

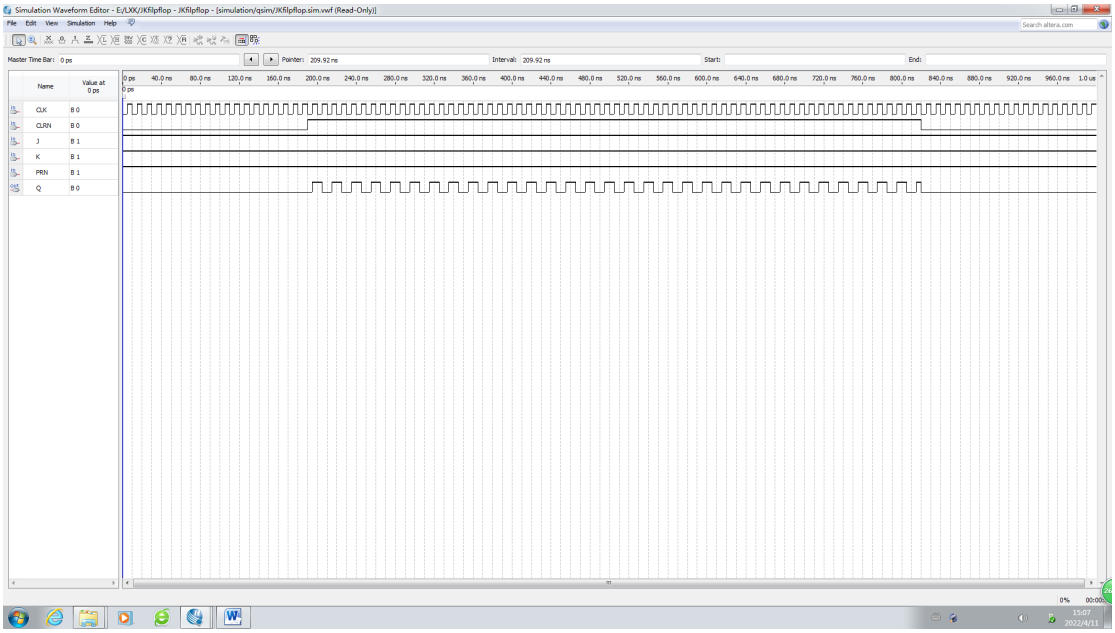
Here is the truth table of JKFF

J	K	Q_t	Q_{t+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Here is the transition table of JKFF

J	K	$Q_t \rightarrow Q_{t+1}$
0	d	0→0
1	d	0→1
d	0	1→1
d	1	1→0

PART2 Asynchronous Trigger flip-flop



Check with the truth table:

Point 1:at 235.0ns

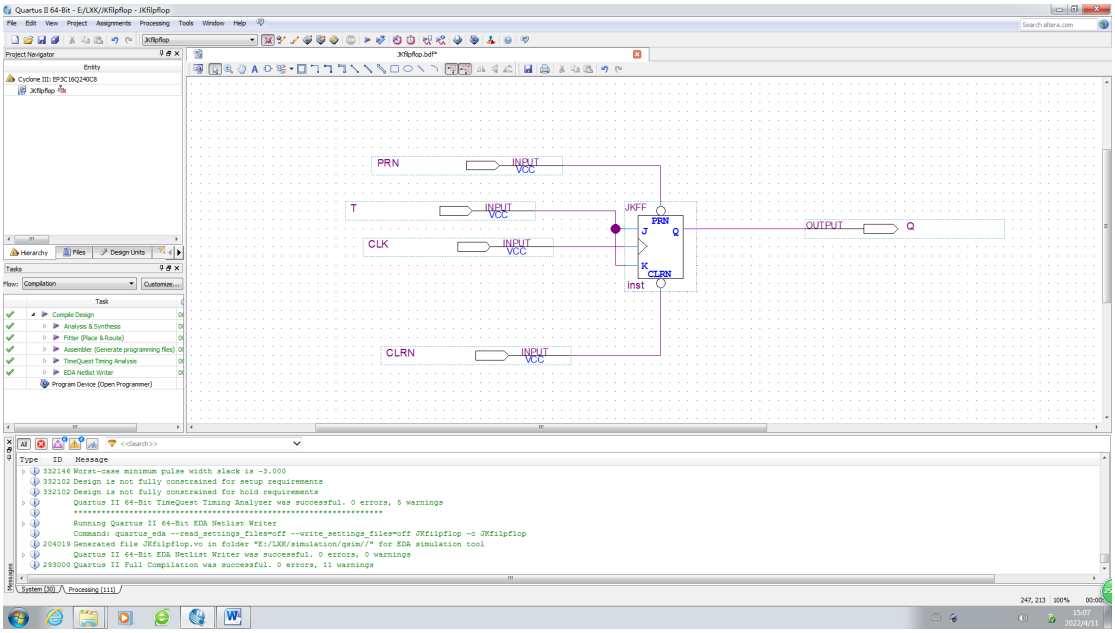
CLK 0⟶1　J=1　K=1　Q 0⟶1 Correct

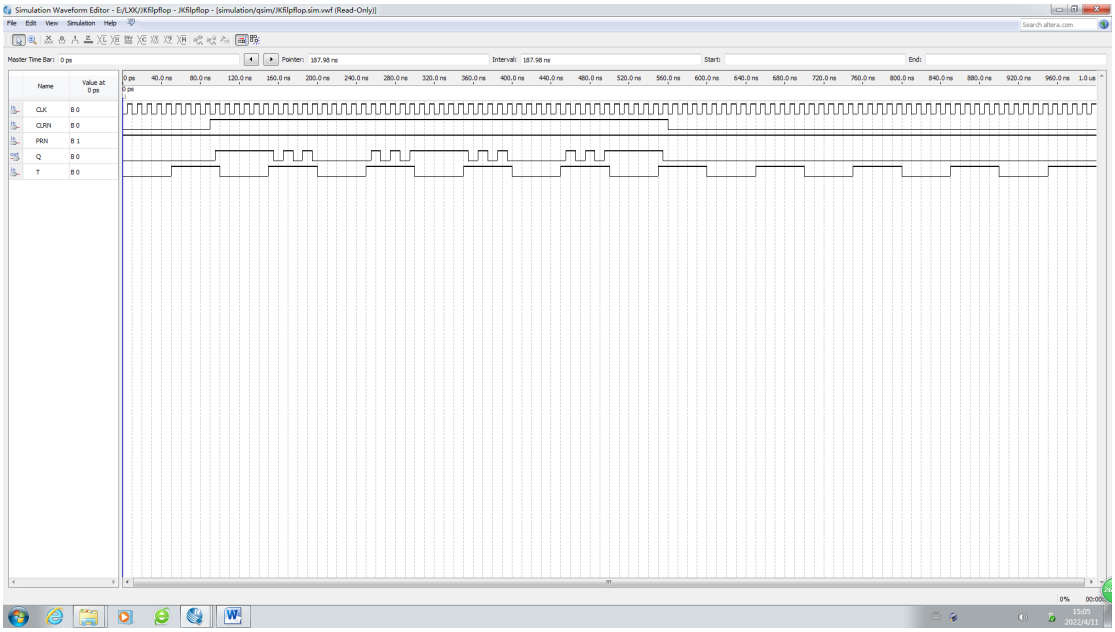
Point 2:at 275.0ns

CLK 0⟶1　J=1　K=1　Q 0⟶1 Correct

……

PART 3 Synchronous Trigger flip-flop





Check with the truth table

Point 1: at 155.0ns

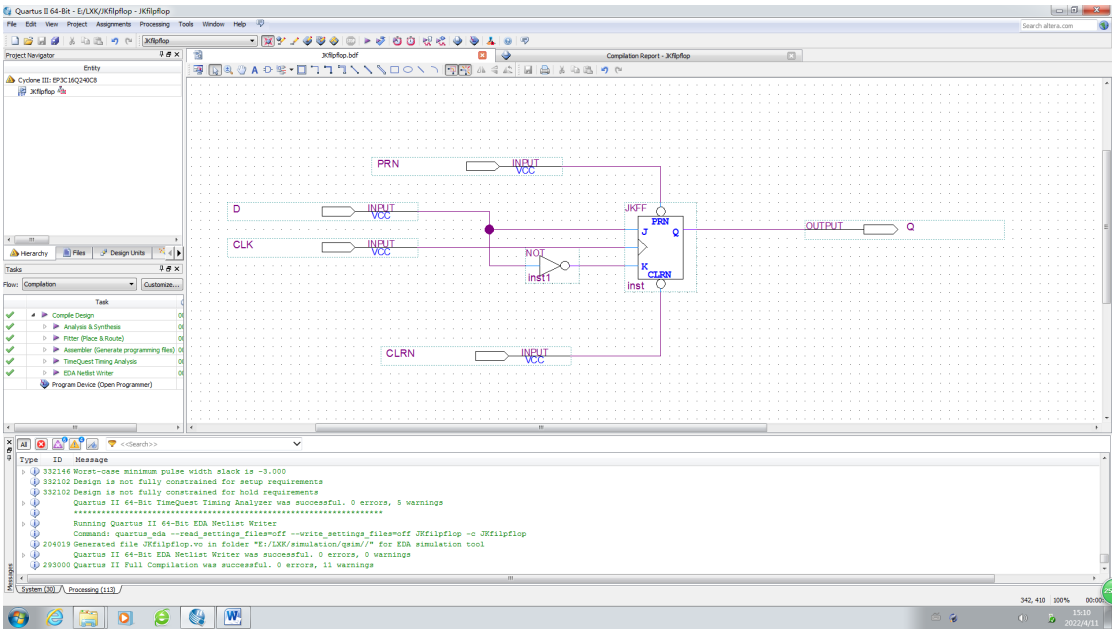
CLK 0⟶1　T=1　Q 1⟶0　Correct

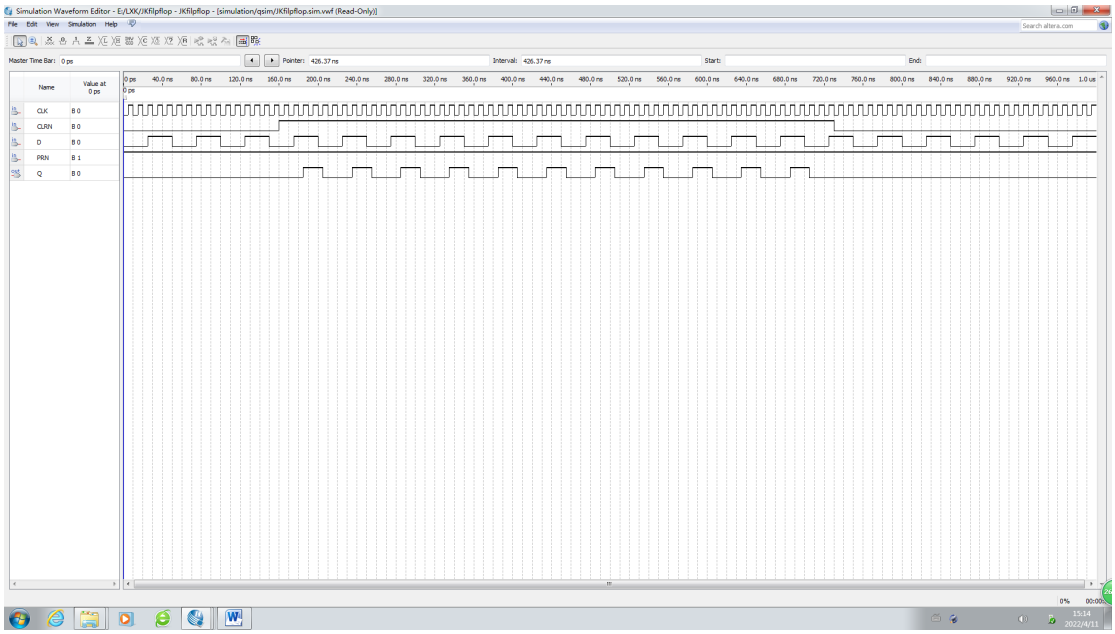
Point 2: at 285.0ns

CLK 0⟶1　T=1　Q 1⟶0　Correct

……

PART 4 Delay flip-flop





Check with the truth table

Point 1: at 235.0ns

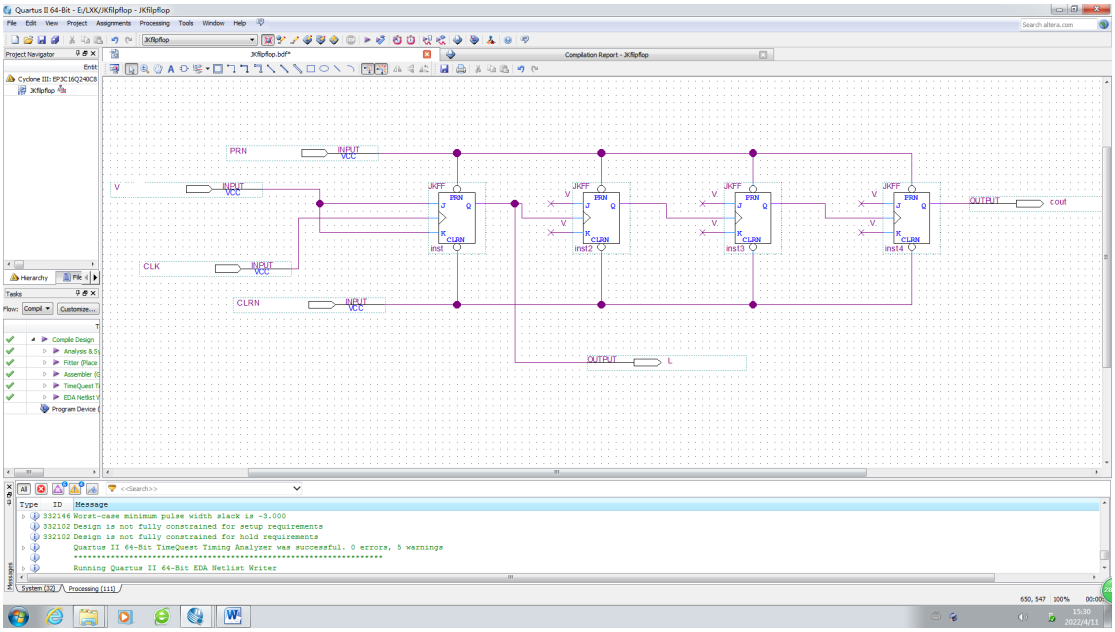
CLK 0⟶1　D=1　Q 0⟶1　Correct

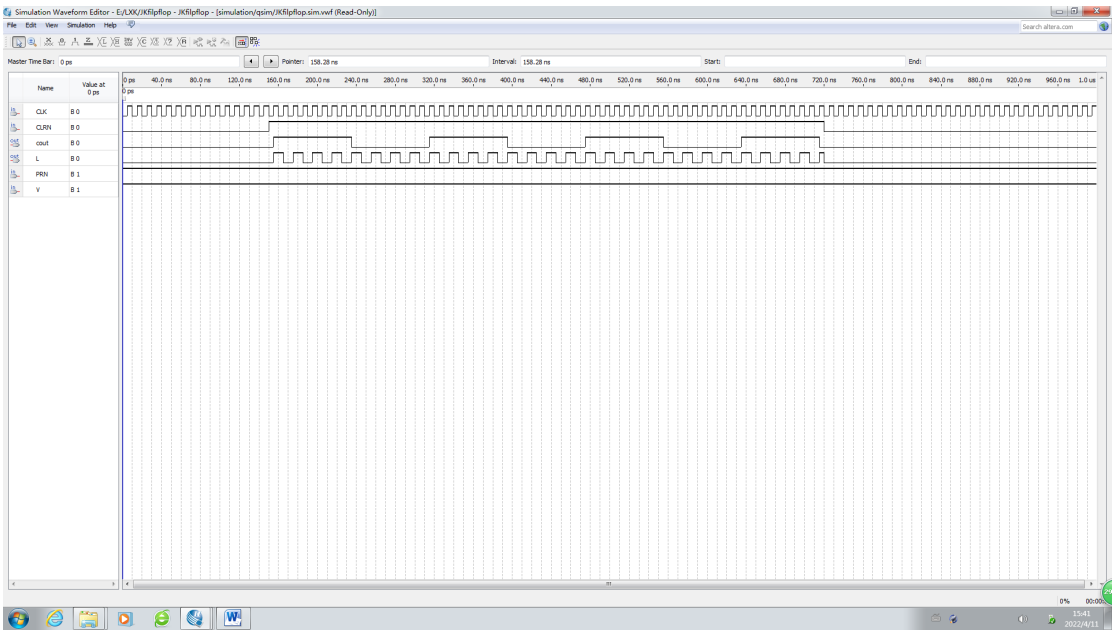
Point 2: at 255.0ns

CLK 0⟶1　D=0　Q 1⟶0　Correct

……

PART 5 4-bit binary counter





Check with the truth table

Point 1: at 160.0ns

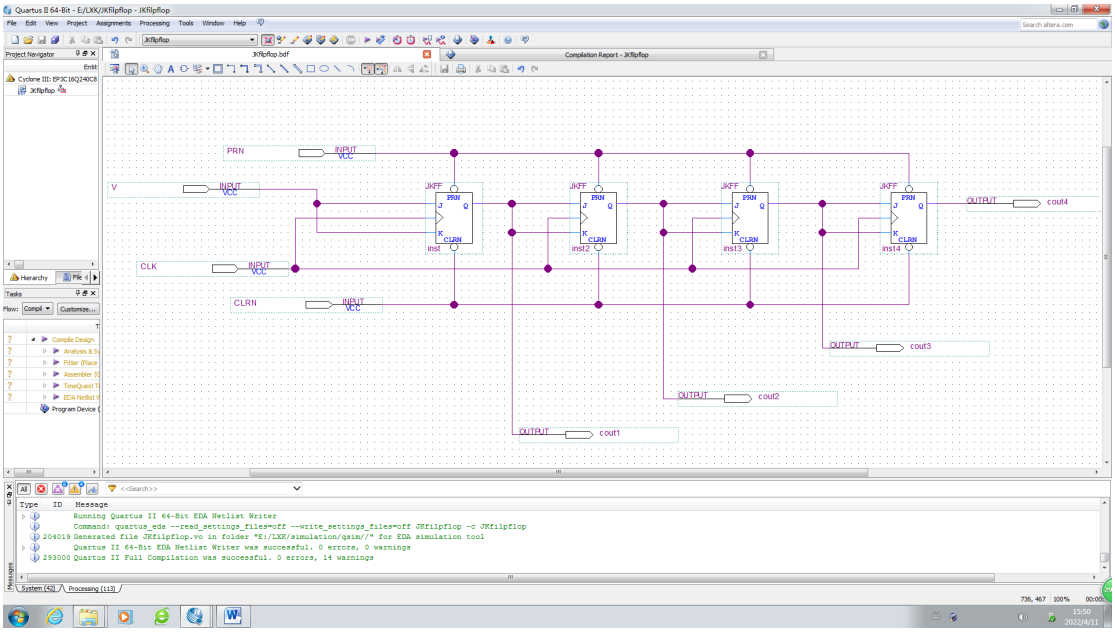
CLK 1→0　V=1　cout=1　Q=1　Correct

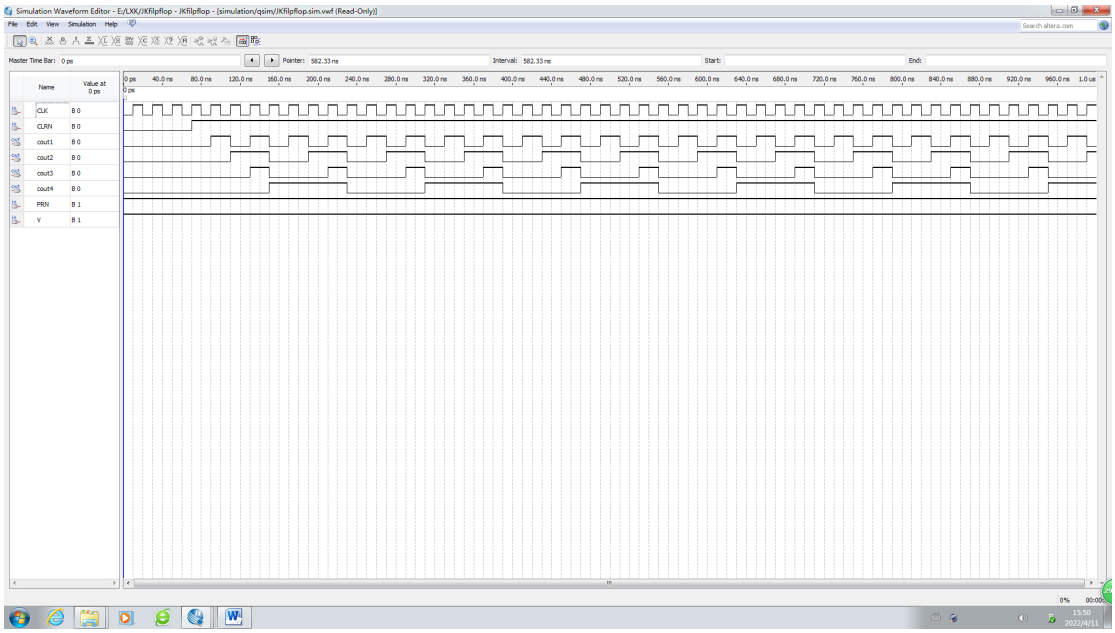
Point 2: at 240.0ns

CLK 1→0　V=1　cout=0　Q=1　Correct

……

PART 6 Synchronous TFF binary counter





Check with the truth table

Point 1: at 120.0ns

CLK 1→0 V=1 cout1=0 cout2=1 cout3=0 cout4=1 Correct

Point 2: at 160.0ns

CLK 1→0 V=1 cout1=0 cout2=0 cout3=0 cout4=1 Correct

.....

Conclusion:

In this seminar, I further learned JKFF and understand the function of there 5 pins. I use JKFF to create many other flip-flops. In part 1, I create a simple JKFF symbol, This flip-flop solves the problem that SRFF cannot be input to 1 at the same time, Master-Slave J-K flip-flop is the very common and popular trigger. In part 2, I need set J and K to logical 1 at all the time, the wave of Q will change when encounter the clock falling edge. This trigger can simply realize the function of counting. In part 3, I connect J and K together to achieve the function of synchronous. When T is 0, the Q is not changed. But when T is set to 1, Q is changed, so T may act as a switch. In part 4, I need add a not gate to TFF. The input is when J=D,K=not D. This device is called a DFF because the next output Q_{t+1} always takes the value of the input D, if the clock is stopped the DFF will store its last input. So the DFF can be made to the shift register. In part 5, I use 4 JKFF to create a 4-bit binary counter, because the clock signal of every JKFF is not the same, so this counter is called asynchronous counter. In part 6, this device require all the clock together and the initial input J and K are equal. The output of the previous JKFF is used

as the input of the next JKFF.

In this seminar, I also find there are some difficult things. At the beginning, I don't know what is PRN and CLRN, even if I know there specific name is "preset" and "clear". Under the guidance of my teacher, I know that at the beginning, CLRN is set to 0 because the previous data needs to be clear. And when I create a waveform, I find that outputs are changed at rising edges, so after class I look up some information and find that hardware has two types, one is rising edge trigger and the other is falling edge trigger.