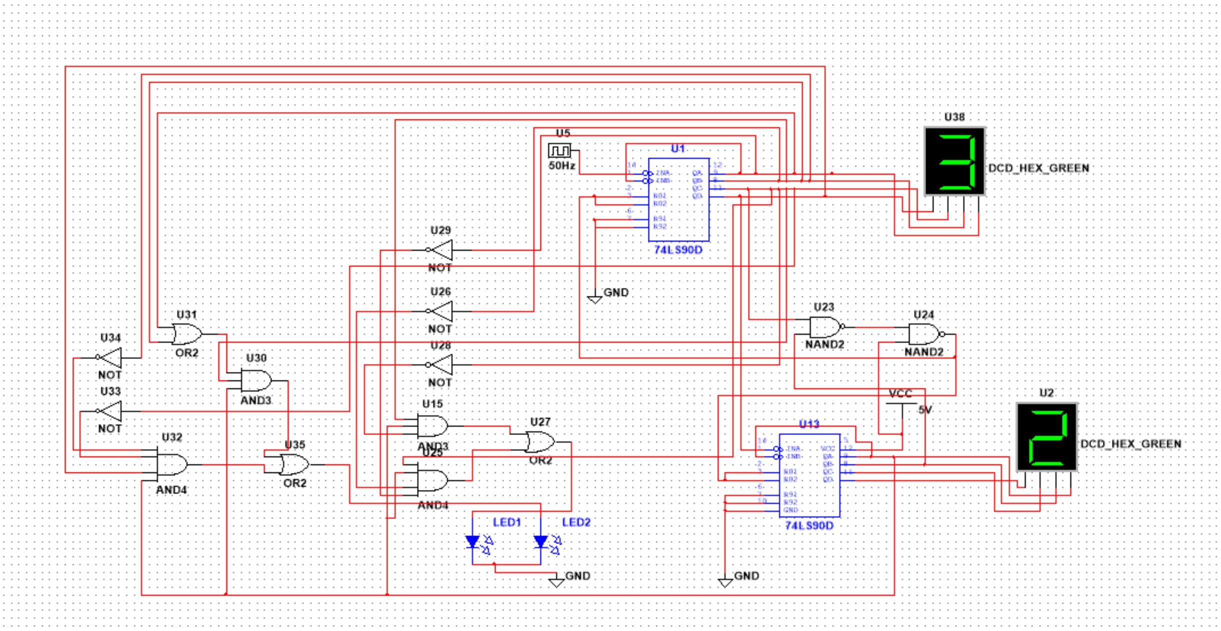
Digital Circuit Project -- Counter

An Lihua 2018531021 SIST

I Multisim Simulation

In Multisim Simulation, I design a ciruit to realise a counter count to the addition of each bit of my student number, which is 2+0+1+8+5+3+1+0+2+1=23. The circuit is given below and the file will be attached to the report.

In this part, I use a component 74LS90D which has the counter's function, it has a input and four bit output. The 7-segment display I choose is DCD_HEX which already includes the decoder part. The two counters 74LS90D consist a 23 base counter,connected by two NAND gate. For the led, I need to light the red led at 12,13,14, so I design a logic combination to input the signal of 12,13,14. The same as the yellow led. However, as the circuit is too complex, there is a delay on some signal, we can see at 16,18 the led go wrong while the designed logic is completely right.



II VHDL Programming

In this part, I realise the 23 base counter (in fact 24 base counter in order to show the number 23) by VHDL programming. The code include two sources, one is the design source which design the 23 base counter; the other is the simulation source which generate the clock , counter enable and reset signals. The two parts of code is given below and the orgin file will be attached to the report.

The Design Source: Counter

In [ ]:

```
Library ieee;
```

```
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;

entity counter23 is
port
  (RST,CE,CLK :in std_logic;
  Q:out std_logic_vector(5 downto 0);
  F:out std_logic
  );
end entity;

architecture rtl of counter23 is
signal tmp  :std_logic_vector(5 downto 0);
begin
  process(clk)
    begin
      if(clk'event and clk='1') then
        if(rst='0')then
          tmp<="000000";
          f<='0';
        elsif(ce='1') then
          if(tmp="010111")then
            tmp<="000000";
          else
            tmp<=unsigned(tmp)+'1';
          end if;
        end if;
      end if;
      q<=tmp;
      if(tmp="010111")then f<='1';
      elsif(tmp="000000")then f<='0';
      end if;
  end process;
end rtl;
```

The Simulation Source: Testbench

In [ ]:

```
library ieee;
use ieee.std_logic_1164.all;

entity counter is
end counter;

architecture rtl of counter is
  component counter23
    port(
      RST,CE,CLK :in std_logic;
      Q:out std_logic_vector(5 downto 0);
      F:out std_logic
      );
  end component;
  signal rst  :std_logic:='0';
  signal ce   :std_logic:='0';
  signal clk  :std_logic:='0';
  signal f    :std_logic:='0';
  signal  q   :std_logic_vector(5 downto 0);
      constant clk_period :time :=20 ns;
  begin
```

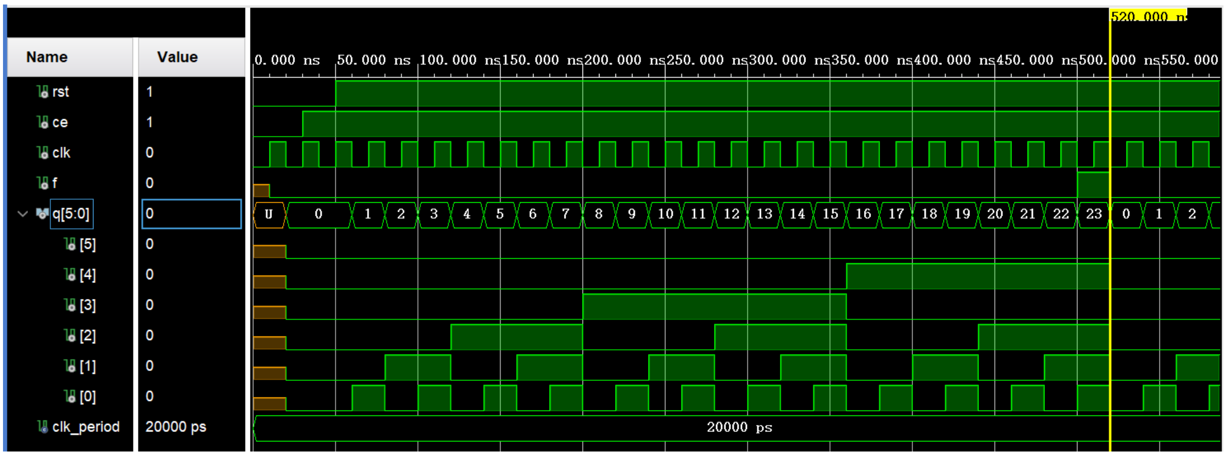
```
    instant:counter23 port map
    (
      clk=>clk,ce=>ce,rst=>rst,q=>q,f=>f
      );

  clk_gen:process
  begin
    wait for clk_period/2;
    clk<='1';
    wait for clk_period/2;
    clk<='0';
  end process;

  rst_gen:process
  begin
    rst<='0';
    wait for 50 ns;
    rst<='1';
    wait;
  end process;

  ce_gen:process
  begin
    ce<='0';
    wait for 30ns;
    ce<='1';
    wait;
  end process;
end rtl;
```

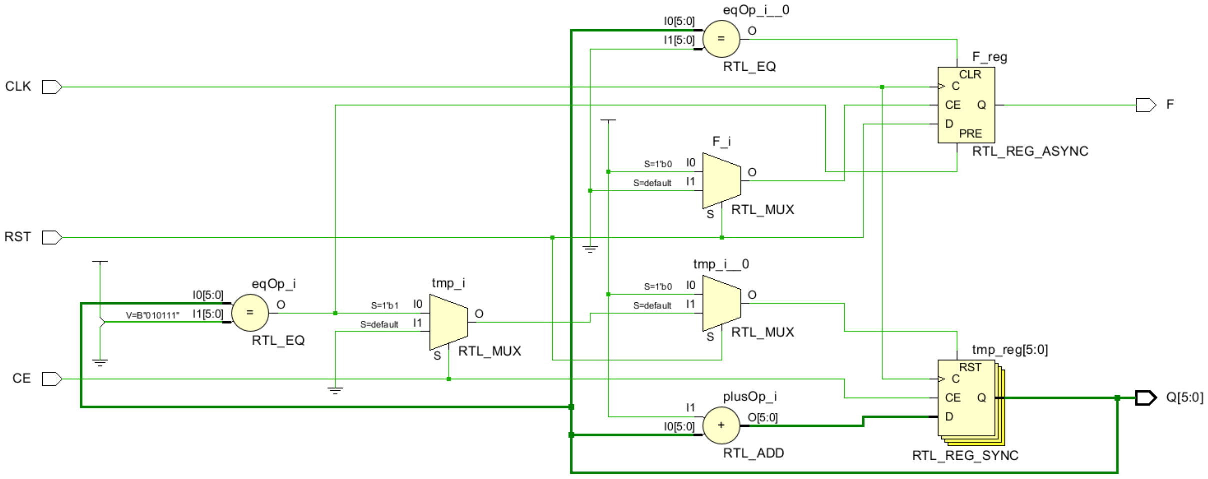
And the simulation waveform is:

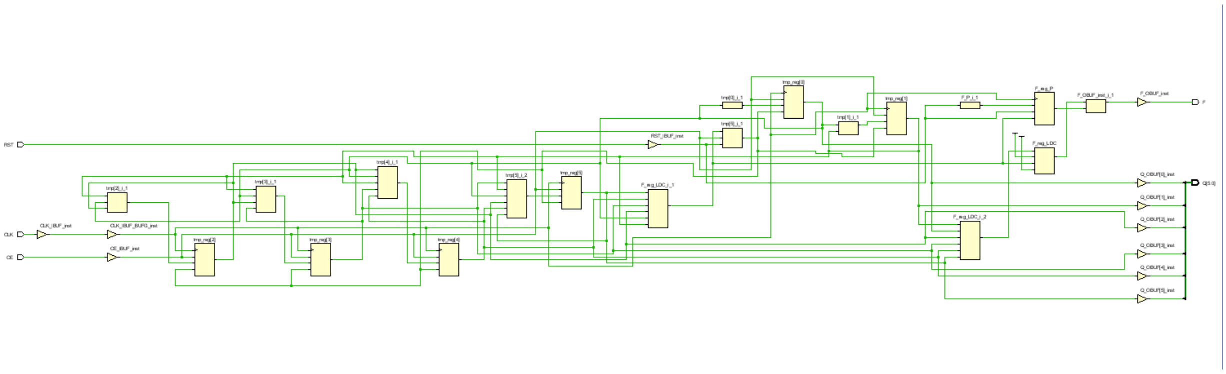


Name	Value
rst	1
ce	1
clk	0
f	0
q[5:0]	0
[5]	0
[4]	0
[3]	0
[2]	0
[1]	0
[0]	0
clk_period	20000 ps

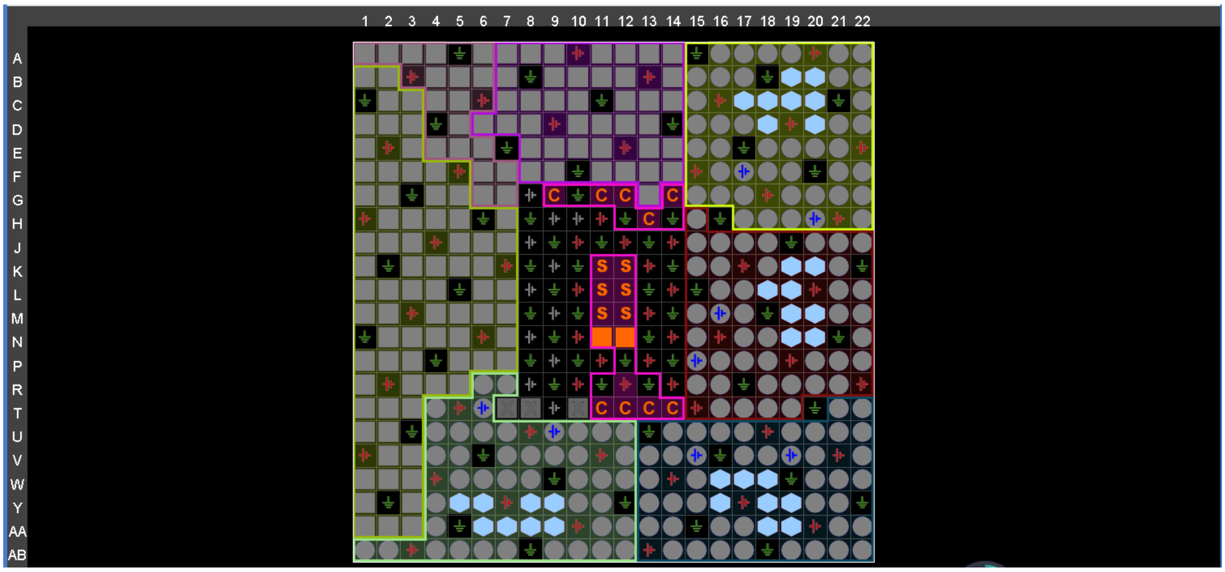
In the figure above, rst meas RESET, clk means CLOCK, ce means COUNTER ENABLE, f means FLAG, q is the OUTPUT. We can see that the counter count to 23 and the Flag is high at the last bit which means a cycle is over.

Then synthesize and get the circuit diagram in the gate level.





We also generate the device diagram.



IV Conclusion

Compare the Multisim simulation and VHDL programming, we find that in the Multisim part, the circuit we design generally is not the simplest then it will delay and make the output not stable. In the VHDL programming, we only need to write down the logic and don't need to think of the complexity of the circuit.

Through this project, I get a much more profound understanding of digital circuit, especially the structure of counter and the theory of some basic components. I am proficent in basic logic gate by doing this project and have a preliminary understanding of VHDL programming, knowing the basic process of using vivado. These are all the essential skills for futher study.