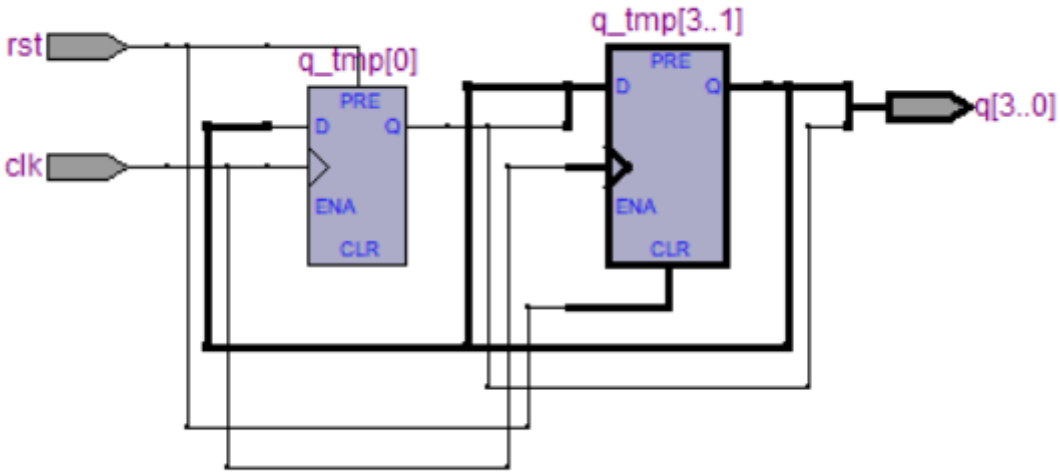
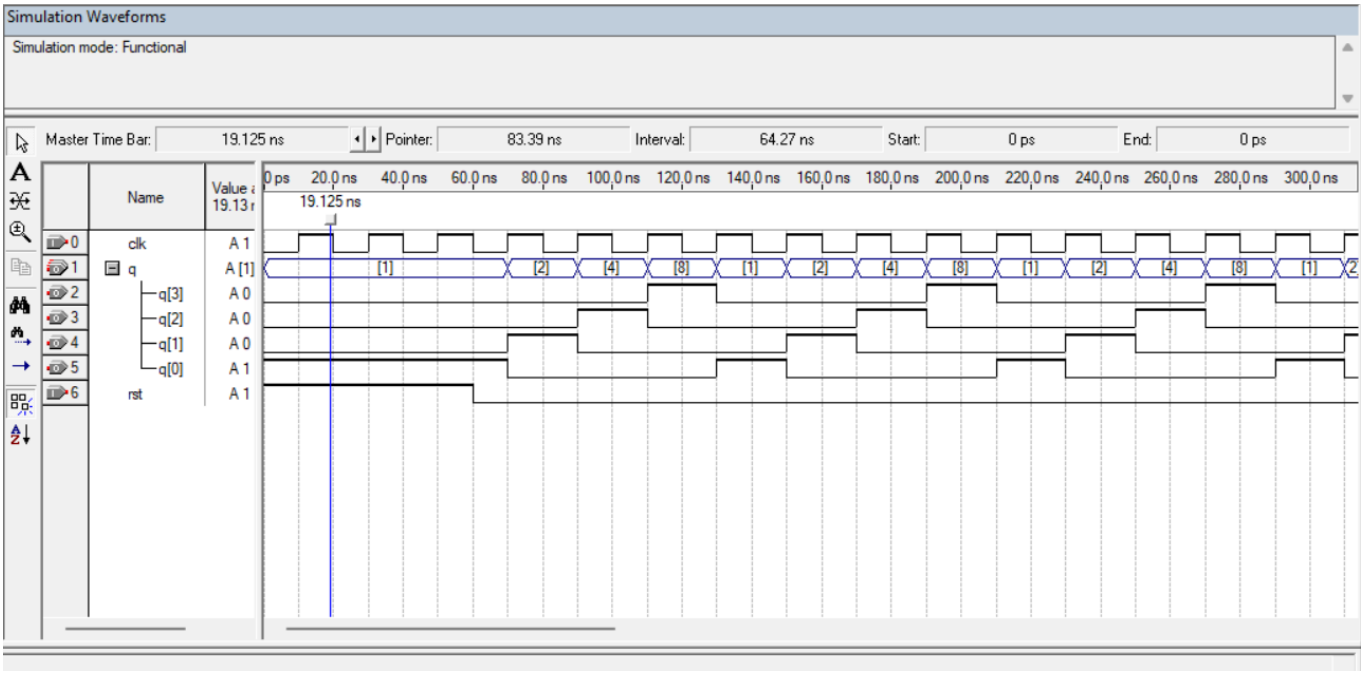
Tutorial 9

Name: Nour-aldin Ibrahim Ahmed Elbadawy
ID: 2018-13394

Ring Counter

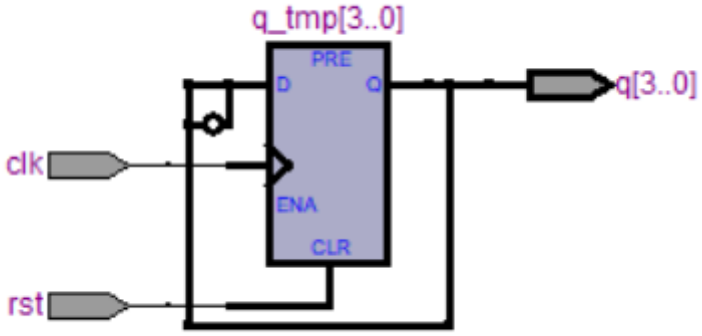
```
library ieee;
use ieee.std_logic_1164.all;

entity ring is
port(
                clk: in std_logic;
                rst: in std_logic;
                q: out STD_LOGIC_VECTOR(3 DOWNTO 0) );

end entity;


architecture behave of ring is
signal q_tmp:std_logic_vector(3 downto 0) :="0000";

begin
        process(clk,rst)
        begin
                if rst = '1' then
                        q_tmp <= "0001";
                elsif rising_edge(clk) then
                        q_tmp(1)<= q_tmp(0);
                        q_tmp(2)<= q_tmp(1);
                        q_tmp(3)<= q_tmp(2);
                        q_tmp(0)<= q_tmp(3);
                end if;
        end process;
q<=q_tmp;
end behave;
```

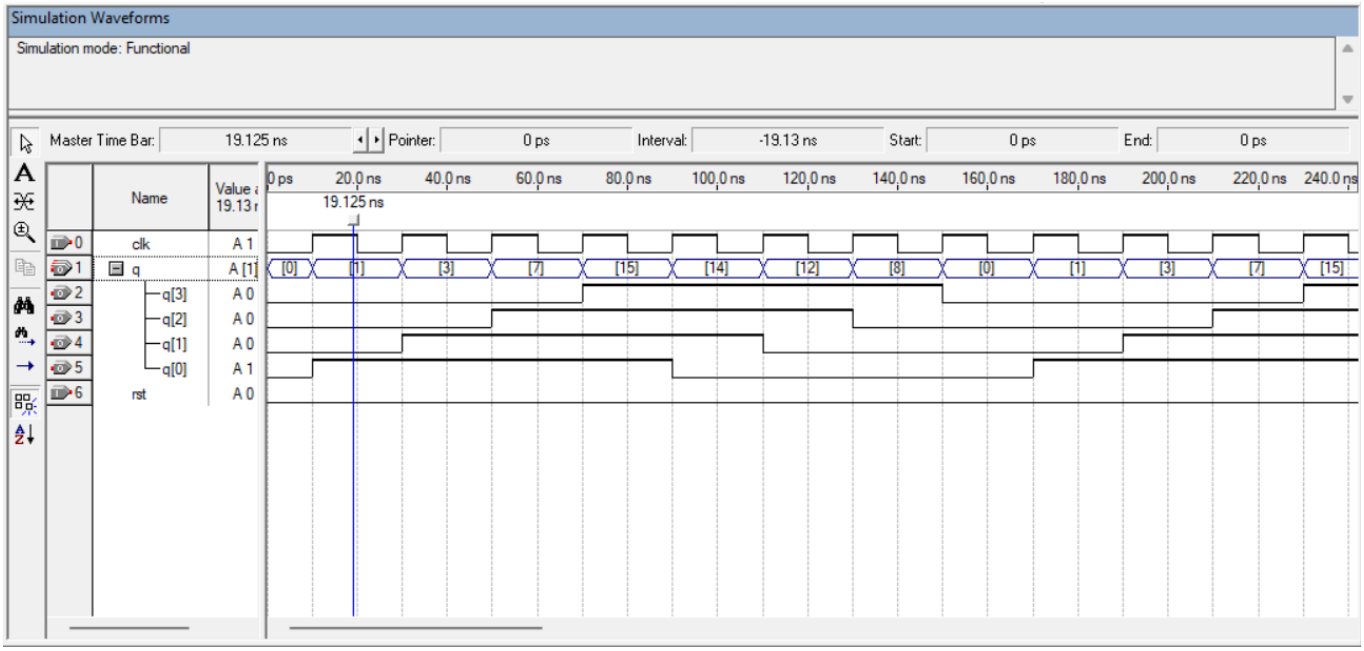

rst
clk
q_tmp[0]
PRE
D
Q
ENA
CLR
q_tmp[3..1]
PRE
D
Q
ENA
CLR
q[3..0]


Simulation Waveforms
Simulation mode: Functional
Master Time Bar: 19.125 ns
Pointer: 83.39 ns
Interval: 64.27 ns
Start: 0 ps
End: 0 ps
Name
clk
q
q[3]
q[2]
q[1]
q[0]
rst
19.125 ns

johnson

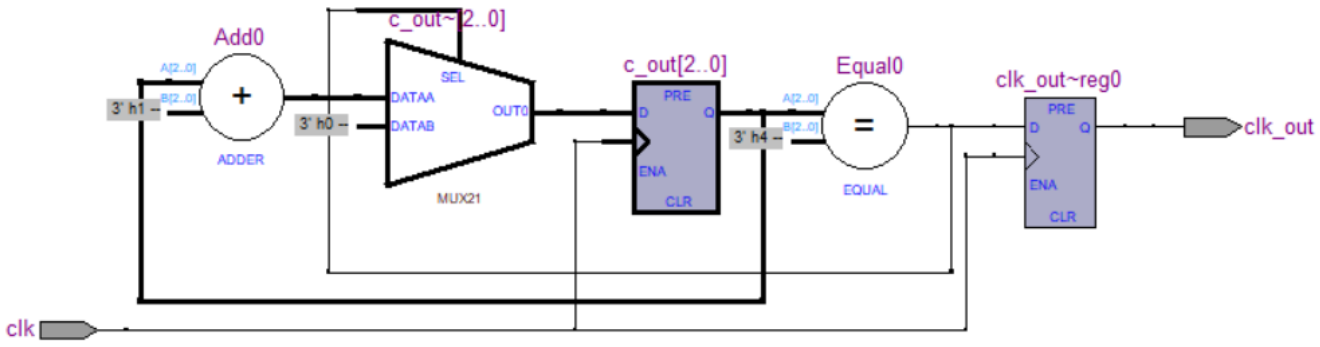
```
library ieee;
use ieee.std_logic_1164.all;

entity johnson is
port(
                clk: in std_logic;
                rst: in std_logic;
                q: out STD_LOGIC_VECTOR(3 DOWNTO 0) );

end entity;


architecture behave of johnson is
signal q_tmp:std_logic_vector(3 downto 0) :="0000";

begin
        process(clk,rst)
        begin
                if rst = '1' then
                        q_tmp <= "0000";
                elsif rising_edge(clk) then
                        q_tmp(1)<= q_tmp(0);
                        q_tmp(2)<= q_tmp(1);
                        q_tmp(3)<= q_tmp(2);
                        q_tmp(0)<= not q_tmp(3);
                end if;
        end process;
q<=q_tmp;
end behave;
```


q_tmp[3..0]
PRE
D
Q
q[3..0]
clk
ENA
CLR
rst


Simulation Waveforms
Simulation mode: Functional
Master Time Bar: 19.125 ns
Pointer: 0 ps
Interval: -19.13 ns
Start: 0 ps
End: 0 ps
Name
clk
q
q[3]
q[2]
q[1]
q[0]
rst
0 ps
20.0 ns
40.0 ns
60.0 ns
80.0 ns
100.0 ns
120.0 ns
140.0 ns
160.0 ns
180.0 ns
200.0 ns
220.0 ns
240.0 ns
19.125 ns

clock divider

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

use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;


entity clock is
port(
                clk: in std_logic;
                clk_out: out STD_logic );

end entity;


architecture behave of clock is
signal q_tmp:std_logic_vector(3 downto 0) :="0000";

begin
        process(clk)
        variable c_out: integer range 0 to 4;
        begin
                if rising_edge(clk) then
                                if(c_out=4)then
                                        c_out:=0;
                                        clk_out<='1';
                                else
                                        c_out:= c_out+1;
                                        clk_out<= '0';
                                end if;
                end if;
        end process;
end behave;
```


Add0
A[2..0]
B[2..0]
3' h1
ADDER
3' h0
c_out~[2..0]
SEL
DATAA
DATAB
OUT0
MUX21
c_out[2..0]
PRE
D
Q
ENA
CLR
A[2..0]
B[2..0]
3' h4
Equal0
EQUAL
clk_out~reg0
PRE
D
Q
ENA
CLR
clk_out
clk


Simulation Waveforms
Simulation mode: Functional
Master Time Bar: 19.125 ns
Pointer: 0 ps
Interval: -19.13 ns
Start:
End:
Name
Value at 19.13 ns
0
clk
A 1
1
..._out
A 0
0 ps
20.0 ns
40.0 ns
60.0 ns
80.0 ns
100.0 ns
120.0 ns
140.0 ns
160.0 ns
180.0 ns
19.125 ns