DIGITAL LOGIC WITH VHDL DESIGN

Dr Tin Thet Nwe

structural description for shift

Q. In digital systems it is often necessary to have circuits that can shift the bits of a vector by one or more bit positions to the left or right. Design a circuit that can shift a four-bit vector $W = w_3w_2w_1w_0$ one bit position to the right when a control signal Shift is equal to 1. Let the outputs of the circuit be a four-bit vector $Y = y_3y_2y_1y_0$ and a signal k.

Use

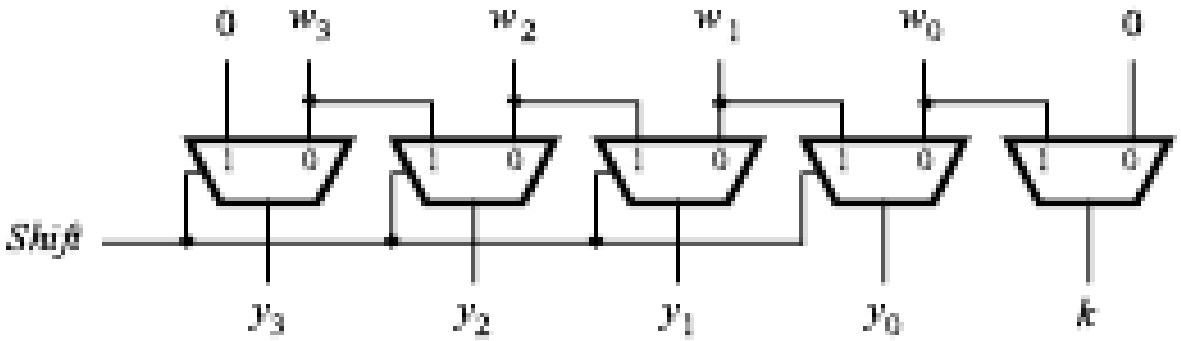


Figure 6.55 A shifter circuit.

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity mux2To1 is
(    a,b,s: in std_logic;
     c:     out std_logic;
);
end mux2To1;
architecture logic_function of mux2To1 is
signal w1,w2,w3: std_logic;
begin
     w1<= not s;
     w2 <= w1 and a ;
    w3<=  s and b;
     c <= w2 or w3;
end logic_function;
```

```
library ieee;
use ieee.std_logic_1164.all;
ENTITY shifter IS
PORT (w : IN STD LOGIC VECTOR(3DOWNTO0) ;
            Shift : IN STD LOGIC ;
            y : OUT STD LOGIC VECTOR(3DOWNTO0) ;
            k : OUT STD LOGIC) ;
END   shifter ;
ARCHITECTURE   Behavior OF shifter IS
BEGIN
mux1:entity work.mux2To1
port map ( a=>'0', b=>w(0) ,s=>shift,c => k );
mux2:entity work.mux2To1
port map ( a=>w(0), b=> w(1),s=>shift,c =>y(0)  );
mux3:entity work.mux2To1
port map ( a=>w(1), b=>w(2) ,s=>shift,c =>y(1)  );
mux4:entity work.mux2To1
port map ( a=>w(2), b=>w(3) ,s=>shift,c =>y(2)  );
mux4:entity work.mux2To1
port map ( a=>w(3), b='0' ,s=>shift,c =>y(3)  );
END Behavior ;
```

```
library ieee;
use ieee.std_logic_1164.all;
ENTITY shifter IS
PORT (w : IN STD LOGIC VECTOR(3DOWNTO0) ;
        Shift : IN STD LOGIC ;
        y : OUT STD LOGIC VECTOR(3DOWNTO0) ;
        k : OUT STD LOGIC) ;
END shifter ;
ARCHITECTURE Behavior OF shifter IS
BEGIN
PROCESS (Shift, w)
BEGIN
        IF      Shift='1'  THEN
        y(3)<='0' ;
        y(2 DOWNTO0)<=w(3DOWNTO1) ;
        k<w(0) ;
ELSE
y<w;
k<'0' ;
ENDIF ;
ENDPROCESS ;
ENDBehavior ;
```

Structural description

Q. Design a circuit that can shift a four-bit vector W =$w_3w_2w_1w_0$ one bit position to the right when a control signal Shift is equal to 1. Let the outputs of the circuit be a four-bit vector Y = $y_3y_2y_1y_0$ and a signal k. Use structural description.

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY shifter IS
    PORT ( w      : IN  STD_LOGIC_VECTOR(3 DOWNTO 0) ;
           Shift  : IN  STD_LOGIC ;
           y      : OUT STD_LOGIC_VECTOR(3 DOWNTO 0) ;
           k      : OUT STD_LOGIC ) ;
END shifter ;

ARCHITECTURE Behavior OF shifter IS
BEGIN
    PROCESS (Shift, w)
    BEGIN
        IF Shift = '1' THEN
            y(3) <= '0' ;
            y(2 DOWNTO 0) <= w(3 DOWNTO 1) ;
            k <= w(0) ;
        ELSE
            y <= w ;
            k <= '0' ;
        END IF ;
    END PROCESS ;
END Behavior ;
```

Figure 6.58 Structural VHDL code that specifies the shifter circuit in Figure 6.55.

Behavioral description

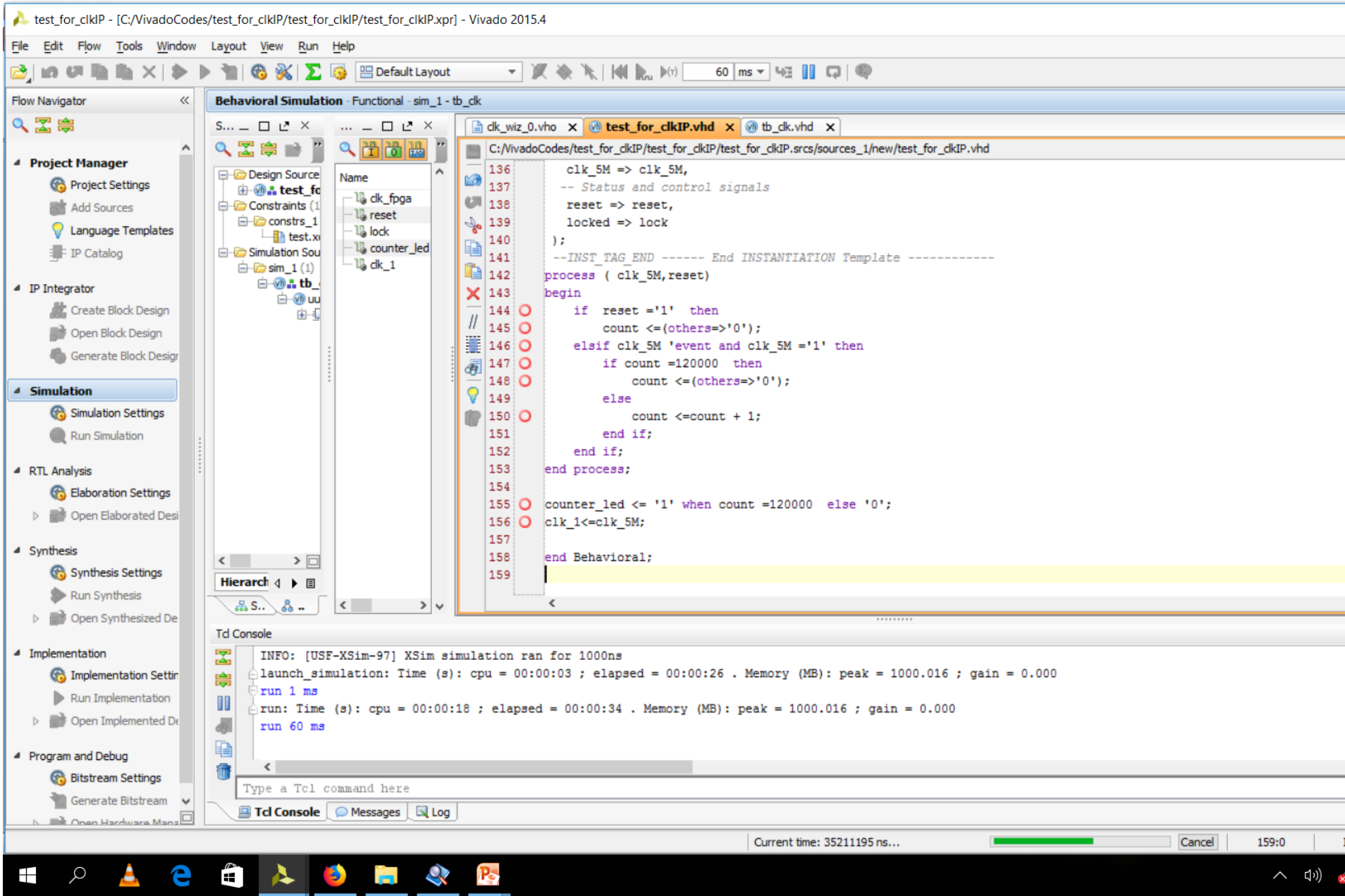
```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;
USE ieee.numeric_std.all ;

ENTITY shifter IS
    PORT ( w     : IN   UNSIGNED(3 DOWNTO 0) ;
           Shift : IN   STD_LOGIC ;
           y     : OUT  UNSIGNED(3 DOWNTO 0) ;
           k     : OUT  STD_LOGIC ) ;
END shifter ;

ARCHITECTURE Behavior OF shifter IS
BEGIN
    PROCESS (Shift, w)
    BEGIN
        IF Shift = "1" THEN
            y <= w SRL 1 ;
            k <= w(0) ;
        ELSE
            y <= w ;
            k <= "0" ;
        END IF ;
    END PROCESS ;
END Behavior ;
```

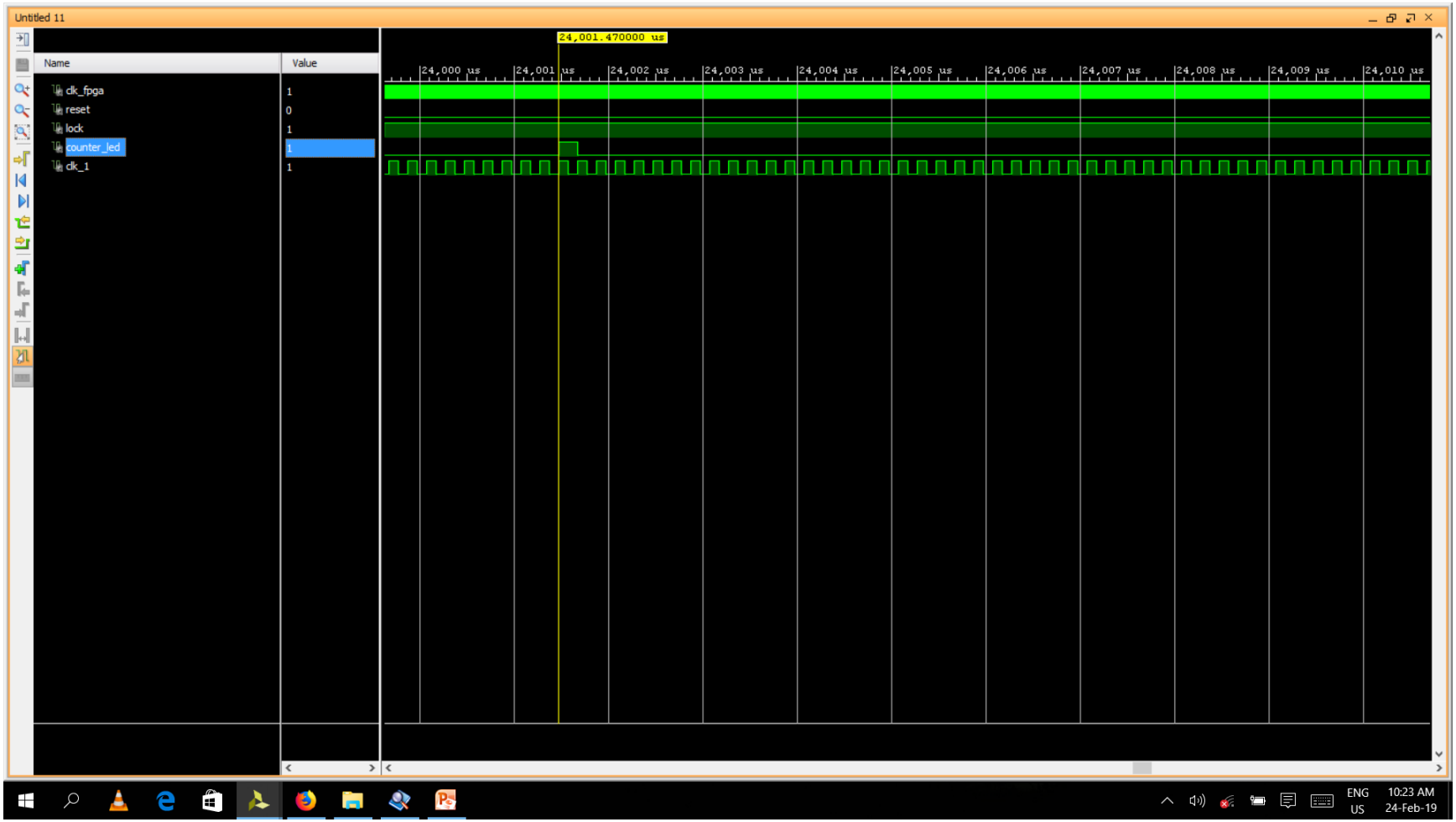
Figure 6.59 Behavioral VHDL code that specifies the shifter circuit in

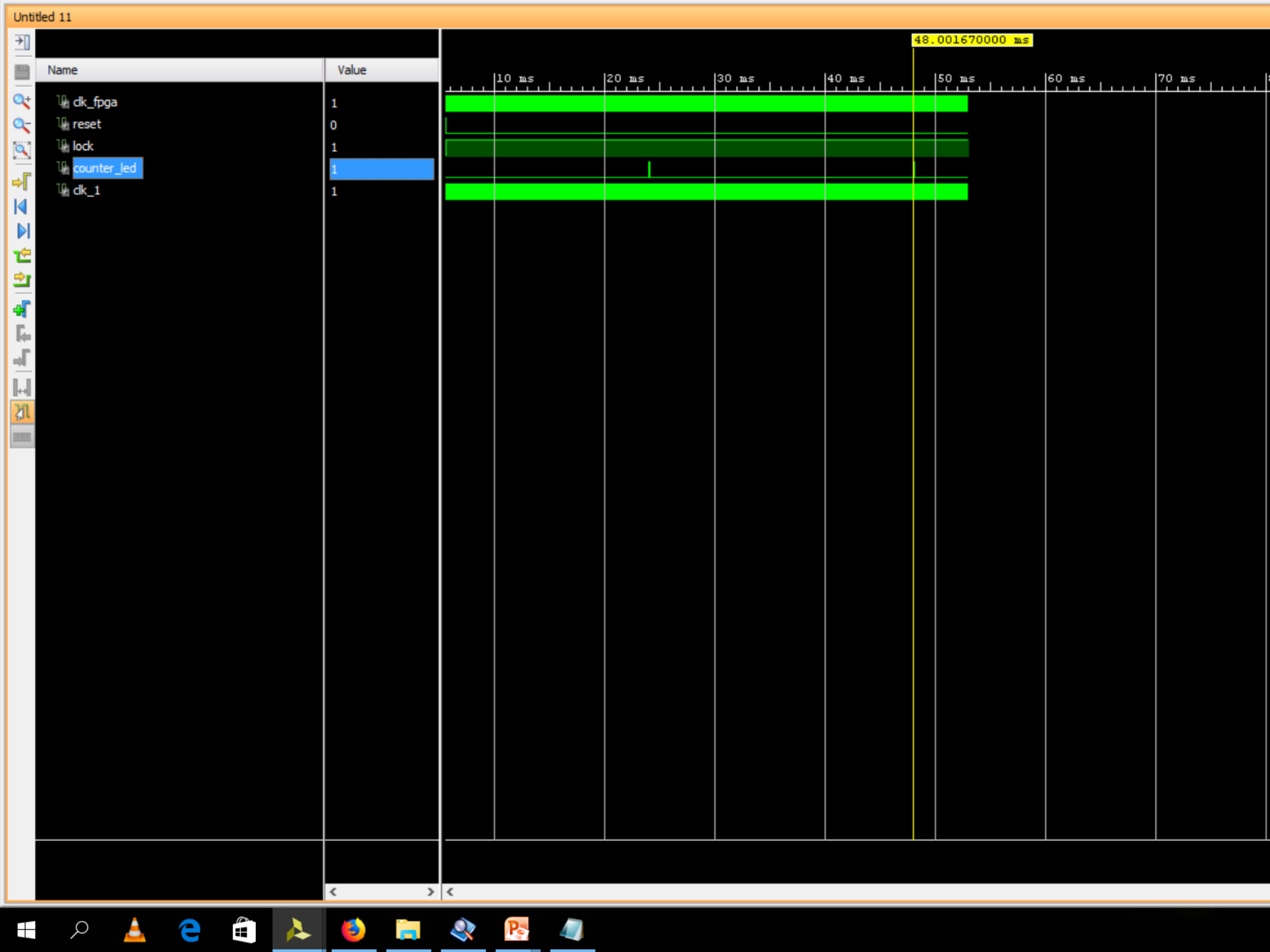
Barrel shifter

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;
USE ieee.numeric_std.all ;

ENTITY barrel IS
    PORT ( w  : IN   UNSIGNED(3 DOWNTO 0) ;
           s  : IN   UNSIGNED(1 DOWNTO 0) ) ;
           y  : OUT  UNSIGNED(3 DOWNTO 0) ) ;
END barrel ;

ARCHITECTURE Behavior OF barrel IS
BEGIN
    PROCESS (s, w)
    BEGIN
        CASE s IS
            WHEN "00" =>
                y <= w ;
            WHEN "01" =>
                y <= w ROR 1 ;
            WHEN "10" =>
                y <= w ROR 2 ;
            WHEN OTHERS =>
                y <= w ROR 3 ;
        END CASE ;
    END PROCESS ;
END Behavior ;
```

Figure 6.60 VHDL code that specifies the barrel shifter circuit in Figure 6.56.


test_for_clkIP - [C:/VivadoCodes/test_for_clkIP/test_for_clkIP/test_for_clkIP.xpr] - Vivado 2015.4
File Edit Flow Tools Window Layout View Run Help
Default Layout
Flow Navigator
Project Manager
Project Settings
Add Sources
Language Templates
IP Catalog
IP Integrator
Create Block Design
Open Block Design
Generate Block Design
Simulation
Simulation Settings
Run Simulation
RTL Analysis
Elaboration Settings
Open Elaborated Design
Synthesis
Synthesis Settings
Run Synthesis
Open Synthesized Design
Implementation
Implementation Settings
Run Implementation
Open Implemented Design
Program and Debug
Bitstream Settings
Generate Bitstream
Behavioral Simulation - Functional - sim_1 - tb_clk
Design Sources
Constraints
Simulation Sources
sim_1 (1)
Name
clk_fpga
reset
lock
counter_led
clk_1
clk_wiz_0.vho
test_for_clkIP.vhd
tb_clk.vhd
C:/VivadoCodes/test_for_clkIP/test_for_clkIP.srcs/sources_1/new/test_for_clkIP.vhd
136 clk_5M => clk_5M,
137 -- Status and control signals
138 reset => reset,
139 locked => lock
140);
141 --INST_TAG_END ------ End INSTANTIATION Template ------------
142 process (clk_5M,reset)
143 begin
144 if reset ='1' then
145 count <=(others=>'0');
146 elsif clk_5M 'event and clk_5M ='1' then
147 if count =120000 then
148 count <=(others=>'0');
149 else
150 count <=count + 1;
151 end if;
152 end if;
153 end process;
154
155 counter_led <= '1' when count =120000 else '0';
156 clk_1<=clk_5M;
157
158 end Behavioral;
159
Tcl Console
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:03 ; elapsed = 00:00:26 . Memory (MB): peak = 1000.016 ; gain = 0.000
run 1 ms
run: Time (s): cpu = 00:00:18 ; elapsed = 00:00:34 . Memory (MB): peak = 1000.016 ; gain = 0.000
run 60 ms
Type a Tcl command here
Tcl Console
Messages
Log
Current time: 35211195 ns...
Cancel
159:0


Untitled 11
24,001.470000 us
Name
Value
clk_fpga
1
reset
0
lock
1
counter_led
1
clk_1
1
24,000 us
24,001 us
24,002 us
24,003 us
24,004 us
24,005 us
24,006 us
24,007 us
24,008 us
24,009 us
24,010 us
ENG
US
10:23 AM
24-Feb-19


Untitled 11
Name
Value
clk_fpga
reset
lock
counter_led
clk_1
1
0
1
1
1
48.001670000 ms
10 ms
20 ms
30 ms
40 ms
50 ms
60 ms
70 ms