طراحی سیستم های دیجیتال
تمرین چهارم
مهندسی کامپیوتر، دانشگاه شهید بهشتی

عباس یزدان مهر
۹۹۲۴۳۰۷۷

پدرام رمضان زاده
۹۹۲۴۳۰۸۵

۸ دی ۱۴۰۱

۱ سوالات کدی

۱.۱

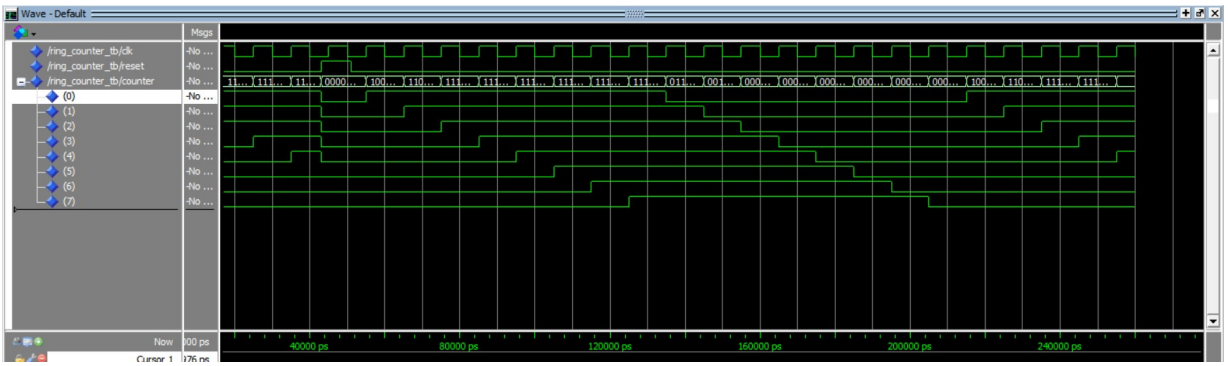
Listing 1: code-part1/ring_counter.vhd

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;

ENTITY ring_counter IS
    GENERIC (N : INTEGER := 4);
    PORT (
        clk : IN STD_LOGIC;
        reset : IN STD_LOGIC;
        counter : OUT STD_LOGIC_VECTOR(0 TO n - 1)
    );
END ring_counter;

ARCHITECTURE beh OF ring_counter IS

    SIGNAL notend : STD_LOGIC;
    SIGNAL temp : STD_LOGIC_VECTOR(0 TO n - 1) := (0 => '1', OTHERS => '0');

BEGIN
    -- if you can code not structrual, do that :)
    notend <= NOT temp(n - 1);

    temp <= (others=>'0') WHEN reset = '1' ELSE
        (notend & temp(0 TO n - 2)) WHEN clk'event AND clk = '1';
    counter <= temp;
END beh; -- beh
```

Listing 2: code-part1/ring_counter_tb.vhd

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY ring_counter_tb IS
END ring_counter_tb;

ARCHITECTURE beh OF ring_counter_tb IS

    SIGNAL clk : STD_LOGIC := '0';
    SIGNAL reset : STD_LOGIC := '0';
    SIGNAL counter : STD_LOGIC_VECTOR(0 TO 7) := (others => '0');

BEGIN

    uut : ENTITY work.ring_counter GENERIC MAP(n => 8) PORT MAP(clk => clk, reset => reset, counter
        => counter);
    clk <= NOT clk AFTER 5 ns;
    reset <= '1' after 43 ns, '0' after 51 ns;
```

```
END beh; -- beh
```



شکل ۱: شبیه سازی قسمت اول

۲.۱

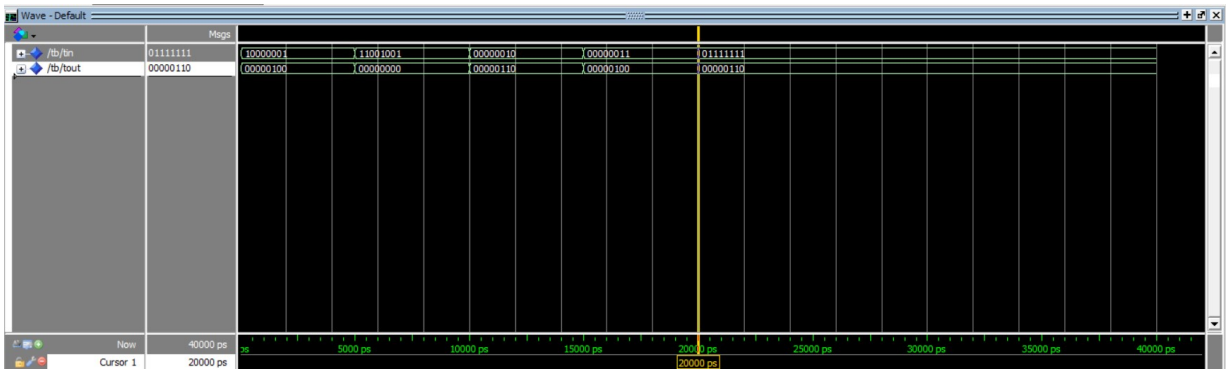
Listing 3: code-part2/zeroonedifference.vhd

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;
USE ieee.std_logic_arith.ALL;

ENTITY zeroonedifference IS
    GENERIC (n : INTEGER := 4);
    PORT (
        inbin : IN STD_LOGIC_VECTOR(0 TO n - 1);
        outbin : OUT STD_LOGIC_VECTOR(0 TO n - 1)
    );
END zeroonedifference;

ARCHITECTURE beh OF zeroonedifference IS

    TYPE int_array IS ARRAY(0 TO n - 1) OF INTEGER;
    TYPE vector_array IS ARRAY(0 TO n - 1) OF STD_LOGIC_VECTOR(0 TO 0);

    SIGNAL notinbin: STD_LOGIC_VECTOR(0 TO n - 1);
    SIGNAL ones : int_array;
    SIGNAL zeros : int_array;
    SIGNAL diff : INTEGER;
    SIGNAL temponefirst : STD_LOGIC_VECTOR(0 TO 0);
    SIGNAL tempones : vector_array;
    SIGNAL tempzerofirst : STD_LOGIC_VECTOR(0 TO 0);
    SIGNAL tempzeros : vector_array;

BEGIN
    temponefirst(0) <= inbin(0);
```

```
    ones(0) <= ieee.numeric_std.to_integer(ieee.numeric_std.unsigned(temponefirst));
    ones_count : FOR i IN 1 TO n - 1 GENERATE
        tempones(i)(0) <= inbin(i);
        ones(i) <= ones(i - 1) + ieee.numeric_std.to_integer(ieee.numeric_std.unsigned(tempones(i)));
    END GENERATE ones_count; -- ones_counts

    notinbin <= NOT inbin;

    tempzerofirst(0) <= notinbin(0);
    zeros(0) <= ieee.numeric_std.to_integer(ieee.numeric_std.unsigned(tempzerofirst));
    zeros_count : FOR i IN 1 TO n - 1 GENERATE
        tempzeros(i)(0) <= notinbin(i);
        zeros(i) <= zeros(i - 1) +
            ieee.numeric_std.to_integer(ieee.numeric_std.unsigned(tempzeros(i)));
    END GENERATE zeros_count; -- zeros_counts

    diff <= ones(n - 1) - zeros(n - 1) WHEN ones(n - 1) > zeros(n - 1) ELSE
        zeros(n - 1) - ones(n - 1);
    outbin <= conv_std_logic_vector(diff, n);
END beh; -- beh
```

Listing 4: code-part2/tb.vhd

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY tb IS
END tb;

ARCHITECTURE beh OF tb IS

    SIGNAL tin : STD_LOGIC_VECTOR(0 TO 7) := "00001001";
    SIGNAL tout : STD_LOGIC_VECTOR(0 TO 7) := "00001001";

BEGIN
    uut : ENTITY work.zeroonedifference GENERIC MAP(n => 8) PORT MAP (inbin => tin, outbin => tout);
    tin <= "10000001", "11001001" AFTER 5 ns, "00000010" AFTER 10 ns, "00000011" AFTER 15 ns,
        "01111111" AFTER 20 ns;
END beh; -- beh
```



شکل ۲: شبیه سازی قسمت دوم

۳.۱

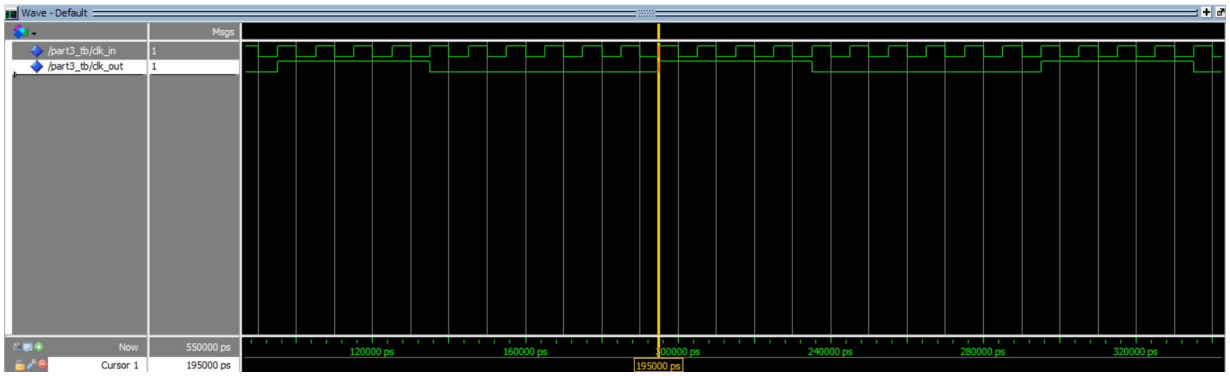
Listing 5: code-part3/part3.vhd

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY part3 IS
    GENERIC (new_duty : INTEGER := 4);
    PORT (
        clk_in : IN STD_LOGIC;
        clk_out : OUT STD_LOGIC
    );
END part3;
ARCHITECTURE behave OF part3 IS

BEGIN
    count : PROCESS (clk_in)
        VARIABLE counter : INTEGER RANGE 0 TO 10;
    BEGIN
        IF (clk_in'event AND clk_in = '1') THEN
            counter := counter + 1;
            IF (counter = new_duty) THEN
                clk_out <= '0';
            END IF;
            IF (counter = 10) THEN
                clk_out <= '1';
                counter := 0;
            END IF;
        END IF;
    END PROCESS;
END behave; -- behave
```

Listing 6: code-part3/part3_tb.vhd

```
LIBRARY ieee;
```

```vhdl
USE ieee.std_logic_1164.ALL;
ENTITY part3_tb IS
END part3_tb;

ARCHITECTURE behave OF part3_tb IS
    SIGNAL clk_in : STD_LOGIC := '0';
    SIGNAL clk_out : STD_LOGIC;
BEGIN
    m1 : ENTITY work.part3 GENERIC MAP (new_duty => 4) PORT MAP (clk_in => clk_in, clk_out =>
        clk_out);
    clk_in <= not clk_in after 5 ns;
END behave; -- behave
```



شکل ۳: شبیه سازی قسمت سوم

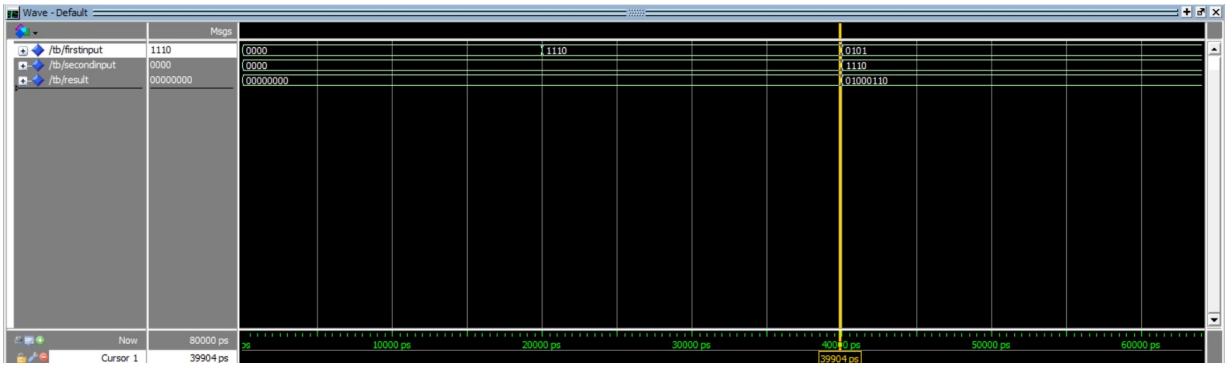
Listing 7: code-part4/mult_usung_add_shift_4b.vhd

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

entity mult_usung_add_shift_4b is
   port(
   firstinput : in std_logic_vector(3 downto 0);
   secondinput: in std_logic_vector(3 downto 0);
   result    : out std_logic_vector(7 downto 0)
      ) ;

end mult_usung_add_shift_4b;

architecture logic_mul of mult_usung_add_shift_4b is
begin
process(firstinput,secondinput)
variable pv,bp :std_logic_vector(7 downto 0);
begin
pv:="00000000";
bp:="0000"& secondinput;
```

```
for i in 0 to 3 loop
   if firstinput(i)='1' then
   pv:=pv+bp;
   end if;
   bp:=bp(6 downto 0)&'0';
end loop;
result<=pv;
end process;
end logic_mul;
```

Listing 8: code-part4/mult_usung_add_shift_4b_tb.vhd

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY tb IS
     END tb;


ARCHITECTURE tbm OF tb IS

  signal firstinput : std_logic_vector(3 downto 0);
  signal secondinput: std_logic_vector(3 downto 0);
  signal result   :  std_logic_vector(7 downto 0);

BEGIN
uut : ENTITY work.mult_usung_add_shift_4b PORT MAP(firstinput => firstinput, secondinput =>
    secondinput, result => result);
  firstinput <= "0000", "1110" after 20 ns, "0101" after 40 ns;
  secondinput <= "0000", "1110" after 40 ns;
end tbm;
```



شکل ۴: شبیه سازی قسمت چهارم

۲ سوالات تحلیلی

۱.۲

اگر جمله ای داخل block Guarded باشد اما Guarded نباشد به طور پیش فرض این سیگنال همیشه درست است.

۲.۲

در کدهای طولانی برای مدیریت بهتر کد و دیباگ کردن راحت تر و همچنین خواناتر شدن کد از block استفاده می کنیم.

۳.۲

```
Type color_type is
Red : std_logic_vector(0 to7);
Blue : std_logic_vector(0 to7);
Black: std_logic_vector(0 to7);
Green : std_logic_vector(0 to7);
End color_type;
--------------------------------------------------------------------------------
type Car is
 record
id_number : std_logic_vector(4 downto 0);
car_owner : std_logic_vector(31 downto 0);
 car_color : color_type;
 end record;
```

پایان