



دانشکده مهندسی کامپیوتر

بسمه تعالی
طراحی خودکار مدارهای دیجیتال

نیمسال دوم ۱۳۹۴

تمرین دوم

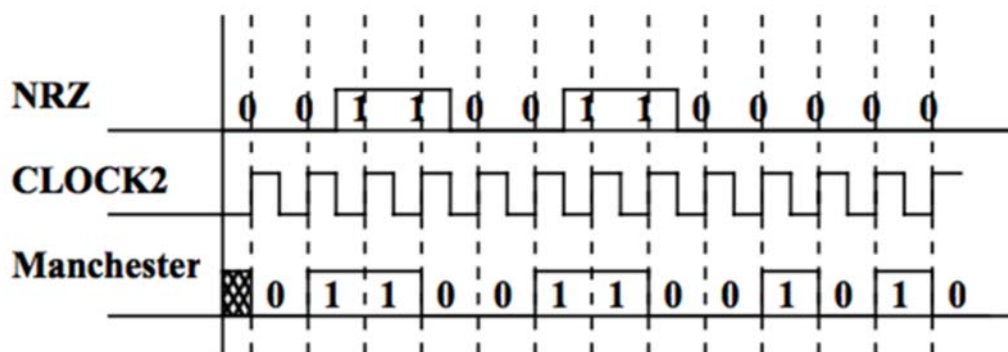


دانشگاه صنعتی امیرکبیر

تاریخ تحویل ۱۳۹۵/۰۲/۲۱

حل سوال ۱ تا ۳ به صورت اختیاری است و نیازی به تحویل آن‌ها نیست (به حل این سوالات نمره‌ای تعلق نمی‌گیرد).

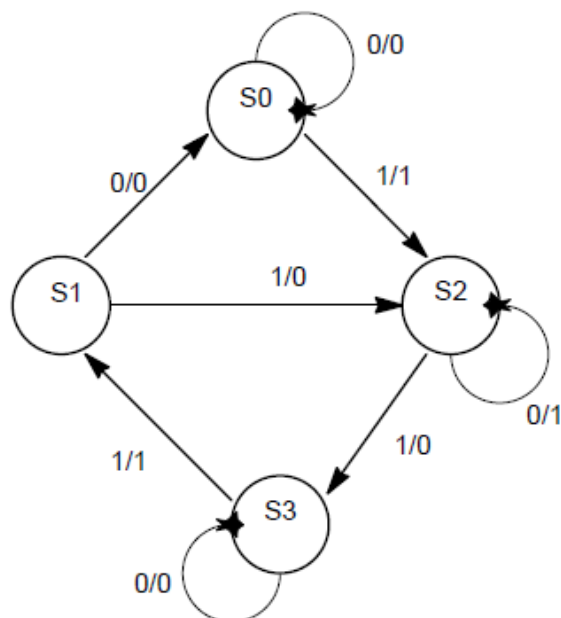
۱. ماشین Moore ای طراحی و با VHDL پیاده‌سازی نمایید که کدینگ NRZ(non return to zero) را به کدینگ Manchester تبدیل نماید.





۲. فرض کنید برای ساختن یک شمارنده با clock-enable از کد زیر استفاده شده است. به نظر شما کدام قسمت می‌تواند مشکل ایجاد کند. چگونه می‌توان کد را تغییر داد به طوری که مشکل گفته شده حل شود همچنین فضای استفاده شده کمتر باشد و latch ناخواسته نداشته باشیم.

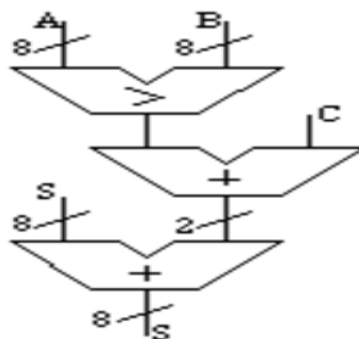
```
enabled_clock <= clock and enable;  
process (enabled_clock)  
begin  
  if RISING_EDGE(enabled_clock) then  
    count <= count + 1;  
  end if;  
end process;
```





تاریخ تحویل ۱۳۹۵/۰۲/۲۱

۳. فرض کنید می‌خواهیم مدار شکل زیر را بسازیم.



برای این کار از کد زیر استفاده می‌کنیم.

```
architecture synth of arithmetic is
  use ieee.std_logic_1164.all;
  use IEEE.numeric_std.all;
  signal a, b : std_logic_vector (7 downto 0);
  signal c : std_logic;
begin
  compare_add: process (a, b, c)
    variable s : Unsigned (7 downto 0);
  begin
    if (a > b) then
      s := s + 1;
    end if;
    if (c = '1') then
      s := s + 1;
    end if;
  end process;
end synth;
```



آیا توصیف بالا باعث ایجاد مدار مورد نظر می شود؟ اگر خیر، اولاً مدار تولید شده را بکشید و سپس کدی ارائه کنید که مدار اولیه را تولید کند. (راهنمایی: مدار اولیه را می توانید بدون استفاده از `if-else` تولید کنید. برای این کار می توان عملگر "<" را `overload` کرد تا به جای `std_logic, boolean` بر گرداند و سپس از نتیجه ی آن برای جمع استفاده می کنیم).

حل سوالات ۴ تا ۸ اجباری بوده عدم تحویل هر کدام موجب کسر نمره خواهد شد.

۴. قطعه کدهای زیر را در نظر گرفته و بگویید که چه اشکالاتی در آن ها وجود دارد و چه اثراتی در عمل کرد هر مدار خواهد داشت. به چه روش هایی می توان آن ها را بر طرف کرد؟

-----1

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity logic1 is
  Port ( A,B,C : in std_logic;
        F : out std_logic);
end logic1;
architecture Behavioral of logic1 is
begin
  process(A,B,C)
  begin
    if A = '1' then
      F <= '1';
    elsif B = '1' and C = '1' then
      F <= '0';
    end if;
  end process;
end Behavioral;
```

-----2



دانشکده مهندسی کامپیوتر

بسمه تعالی

طراحی خودکار مدارهای دیجیتال

نیمسال دوم ۱۳۹۴

تمرین دوم



دانشگاه صنعتی امیرکبیر

تاریخ تحویل ۱۳۹۵/۰۲/۲۱

```
-- next state logic for a FSM
process (state, a, b, c, d, e)
begin
  case state is
    when IDLE =>
      if a = '0' then
        next_state <= INITIAL;
      end if;
    when INITIAL =>
      if a = '1' then
        next_state <= ERROR_FLAG;
      else
        next_state <= SCANNING;
      end if;
    when SCANNING =>
      if b = '1' then
        next_state <= LOCKED;
      elsif b = '0' then
        if c = '0' then
          next_state <= TIME_OUT;
        else
          next_state <= RELEASE;
        end if;
      else
        next_state <= CAPTURE;
      end if;
    when CAPTURE =>
      next_state <= ...
    when LOCKED =>
      next_state <= ...
    when TIME_OUT =>
      next_state <= ...
    when RELEASE =>
      next_state <= ...
    when ERROR_FLAG =>
      next_state <= some_function(a, d, e);
    end case;
  end process;
```



دانشکده مهندسی کامپیوتر

بسمه تعالی
طراحی خودکار مدارهای دیجیتال

نیمسال دوم ۱۳۹۴

تمرین دوم



دانشگاه صنعتی امیرکبیر

تاریخ تحویل ۱۳۹۵/۰۲/۲۱

۵. در کدام قسمت کد زیر transparent latch ایجاد می‌شود؟ کد را به نحوی تغییر دهید که از این مورد جلوگیری کند.

```
process (sel, sel_2, sel_3, a, b)
begin
  if sel = '1' then
    f <= a;
    if sel_2 = '1' then
      g <= not a;
    else
      g <= not b;
      if sel_3 = '1' then
        g <= a xor b;
      end if;
    end if;
  else
    if sel_2 = '1' then
      g <= a and b;
    else
      if sel_3 = '1' then
        g <= a nand b;
      end if;
    end if;
    f <= b;
  end if;
end process;
```



دانشکده مهندسی کامپیوتر

بسمه تعالی
طراحی خودکار مدارهای دیجیتال
نیمسال دوم ۱۳۹۴

تمرین دوم



دانشگاه صنعتی امیرکبیر

تاریخ تحویل ۱۳۹۵/۰۲/۲۱

۶. الف) توضیح دهید سیگنال‌ها و متغیرهای یک پراسس در هنگام سنتز در چه صورتی به حافظه تبدیل می‌شوند.

ب) کد زیر را در نظر بگیرید و بگویید چه تعداد FF تولید خواهد شد.

```
Library IEEE;
use IEEE.Std_Logic_1164.all;
entity COUNTER is
port ( CLK : in std_ulogic;
Q : out integer range 0 to 15 );
end COUNTER;

architecture Behavior of COUNTER is
signal COUNT : integer range 0 to 15 ;
begin

process (CLK)
begin
if CLK`event and CLK = `1` then

if (COUNT >= 9) then
COUNT <= 0;
else
COUNT <= COUNT +1;
end if;

end if;
end process;
Q <= COUNT;
end Behavior;
```

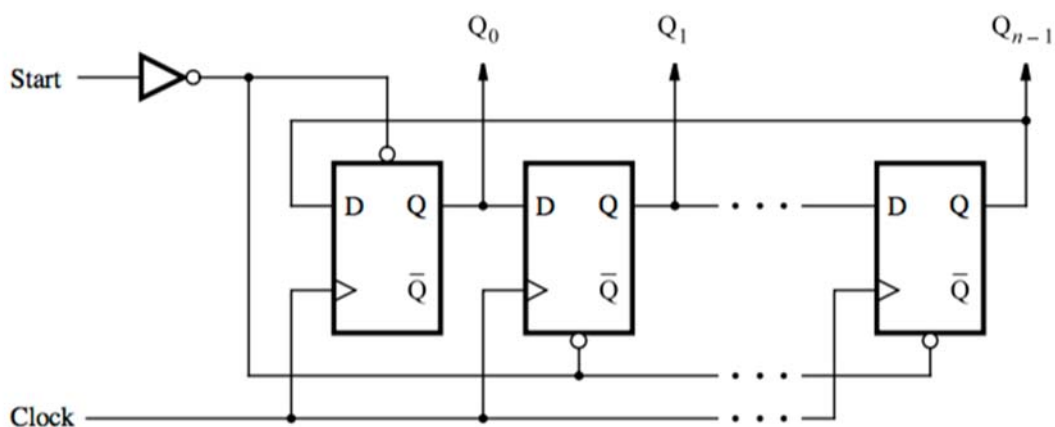


پ) قطعه کد زیر را در نظر بگیرید و تعداد رجیسترهای ساخته شده در هر کدام را بدست آورید (فرض کنید کدها بخشی از یک پراسس هستند که در ابتدای آن، با دستور wait لبه فعال کلاک چک شده است).

```
for I in 0 to 7 loop
  V := V and INPUT(I);
end loop;
```

راهنمایی: برای بررسی دقیق این کد، حلقه را باز کنید (تکرار دستور انتساب به تعداد هشت بار).

۷. با استفاده از generic یک شمارنده ی حلقوی n بیتی (Ring Counter) پیاده سازی کنید (مدار این شمارنده به صورت زیر است ولی شما توصیف آن را در سطح RTL و قابل سنتز بنویسید).





۸. فرض کنید که طراح کد زیر قصد طراحی یک D flip flop با یک ریست active low سنکرون و یک ریست آسنکرون active high را داشته است.
در صورت وجود error در کد زیر آن ها را بیان نمایید و توضیح دهید که چه اثر نامطلوبی بر سیستم می‌گذارد و در نهایت کد را اصلاح نمایید.

```
signal a, b, c, d, e: std_ulogic;  
process(a, b, c, d)  
begin  
    if a = '0' then  
        d <= '0';  
    elsif b = '1' and b'event then  
        d <= c;  
    end if;  
    if e = '1' then  
        d <= '1';  
    end if;  
end process;
```



حل سوال ۹ اختیاری بوده و تحویل آن نمره اضافه خواهد شد.

۹. یک دستگاه خرید خودکار با ۳ نوع سکه کار می‌کند. و در هر بار وارد شدن سکه به دستگاه سیگنال coin_in یکبار یک می‌شود و سیگنال سکه مورد نظر در حالت ۱ و بقیه سکه‌ها در حالت ۰ است. می‌خواهیم ماژولی پیاده کنیم که ابتدا چند سکه از کاربر دریافت کند سپس هنگام یک شدن سیگنال buy_in عملیات خرید را انجام دهد (قیمت کالا از طریق سیگنال price مشخص می‌شود). و بقیه پول کاربر را به او بازگرداند (با حداقل تعداد سکه ممکن). کد VHDL را برای این طرح بنویسید و دنباله بیتی (bitstream) را با هدف حداکثر سرعت کار مدار برای آن تولید کنید.

سپس با استفاده از امکانات ابزارهای سنتز و جایابی و مسیریابی، سعی کنید این طرح را با هدف توان مصرفی کم پیاده‌سازی کنید. نتایج را با (الف) مقایسه کنید و توضیح دهید.

```
entity vending_machine is
port(
    coin_in : in std_logic,
    coin_in_1 : in std_logic,
    coin_in_10 : in std_logic,
    coin_in_100 : in std_logic,
    buy_in : in std_logic,
    price : in std_logic_vector(7 downto 0),
    coin_return : out std_logic,
    coin_return_1 : out std_logic_vector(7 downto 0),
    coin_return_10 : out std_logic_vector(7 downto 0),
    coin_return_100 : out std_logic_vector(7 downto 0));
end entity;
```



دانشکده مهندسی کامپیوتر

بسمه تعالی

طراحی خودکار مدارهای دیجیتال

نیمسال دوم ۱۳۹۴

تمرین دوم



دانشگاه صنعتی امیرکبیر

تاریخ تحویل ۱۳۹۵/۰۲/۲۱
