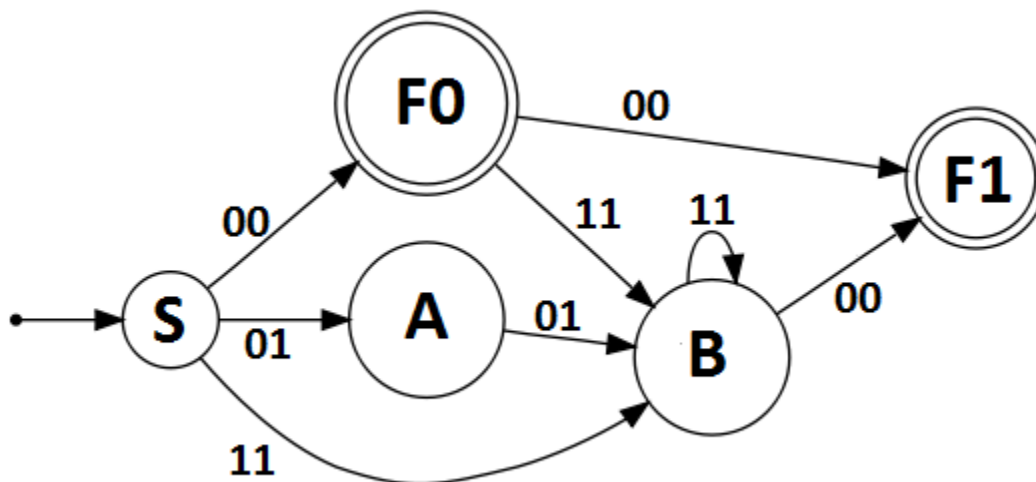


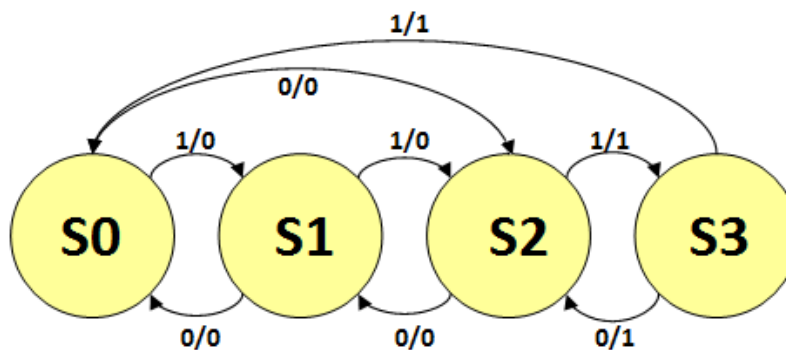


حل سوال ۱ تا ۴ به صورت اختیاری است و نیازی به تحویل آن‌ها نیست (به حل این سوالات نمره‌ای تعلق نمی‌گیرد).

۱. FSM زیر را با کد VHDL توصیف کنید. خروجی سیستم وابسته به State آن است بدین صورت که اگر سیستم در F0 و یا F1 باشد (علامت دو دایره داخل هم نشان‌دهی Final State است) خروجی یک و در غیر این صورت صفر است. در هر State چهار حالت ورودی ممکن است که در گراف زیر همه آن‌ها مشخص نشده است حالت‌هایی که مشخص نشده را Don't care در نظر بگیرید.



۲. FSM زیر را با کد VHDL توصیف کنید. خروجی سیستم وابسته به State و ورودی سیستم است (مشابه ماشین Mealy).





دانشکده مهندسی کامپیوتر

بسمه تعالی  
طراحی خودکار مدارهای دیجیتال  
نیمسال دوم ۱۳۹۴  
تمرین سوم



دانشگاه صنعتی امیرکبیر

تاریخ تحویل ۱۳۹۵/۲/۷

۳. **entity** ارائه شده در شکل زیر را با استفاده از **generic** بازنویسی کنید و با استفاده از دستور **generate**، ۸ بار با مقادیر مختلف آن را **instantiate** کنید.

**entitytest\_core** is

port (

**a\_input** : in **std\_logic\_vector**(31 downto 0);

**b\_input** : in **std\_logic\_vector**(15 downto 0);

**a\_be** : in **std\_logic\_vector**(3 downto 0);

**b\_be** : in **std\_logic\_vector**(1 downto 0);

**clk**, **rst** : in **std\_logic**;

**result** : out **std\_logic\_vector**(31 downto 0);

**valid** : out **std\_logic**

);

end **test\_core**;

۴. برای هر یک از موارد خواسته شده یک **function** بنویسید.

الف) **function** بنویسید که یک **bit\_vector** با هر سائزی را بگیرد و یک **integer** بازگرداند.

ب) **function** بنویسید که یک آرایه یک بعدی از اعداد **integer** را دریافت می کند و میانگین آن ها را محاسبه می نماید. سائز آرایه هر عدد دلخواهی می باشد. درون **function** از هیچ متغیر یا سیگنال اضافه ای نمی توانید استفاده کنید و فقط مجاز به استفاده از ورودی و خروجی آرایه می باشید.

**functionavg\_f** (**A** : in **integer\_vector**) return **integer** is

...

end **function**;

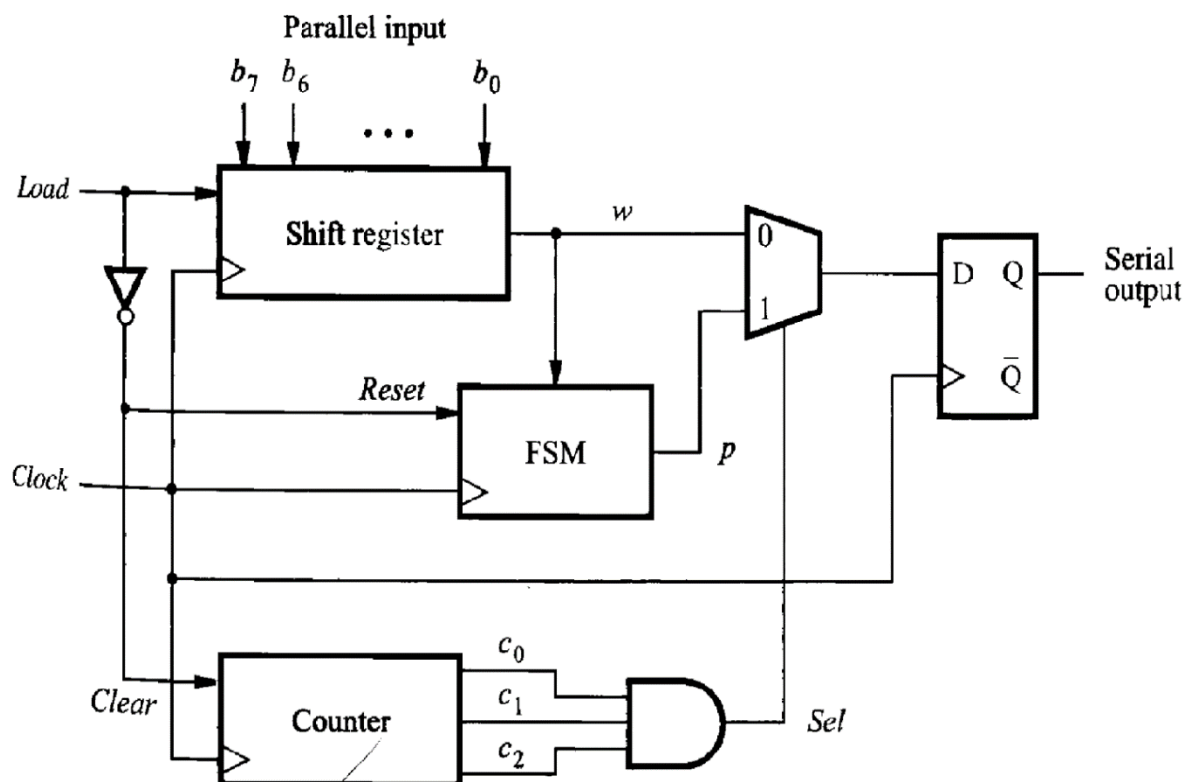


حل سوالات ۵ تا ۱۱ اجباری بوده عدم تحویل هر کدام موجب کسر نمره خواهد شد.

۵. در سیستم‌های کامپیوتری اغلب ارسال داده‌ها به صورت سریال مطلوب است. بنابراین در سمت ارسال کننده باید بتوانیم داده‌ی به صورت موازی را به فرم سریال در آوریم. فرض کنید می‌خواهیم کاراکترهای اسکی را با استفاده از این سیستم ارسال کنیم. این کاراکترها هفت بیتی می‌باشند همینطور از یک بیت parity به عنوان بیت هشتم استفاده می‌کنیم. نمونه‌ای از چنین مداری در شکل (۱) نشان داده شده است. اگر تعداد یک‌ها فرد باشد بیت parity یک و در غیر این صورت صفر خواهد بود. با استفاده از اطلاعات داده شده:

الف) مدار داخلی FSM موجود در شکل را بدست آورید.

ب) کد VHDL بنویسید که عملیات شکل زیر را پیاده‌سازی کند. همه‌ی Component‌های شکل باید پیاده‌سازی شوند.





۶. State Diagram ای برای FSM با ورودی w و خروجی z ارائه کنید. مقدار z برابر یک خواهد شد در صورتی که هر یک از دنباله‌های ۱۱۱۱ و یا ۱۰۰۱ را در ورودی تشخیص دهد، در غیر این صورت مقدار آن برابر صفر خواهد بود (Overlap مجاز است). نمونه‌ای از ورودی و خروجی در شکل زیر مشخص شده است. کد VHDL آن را یک بار با سه پروسس یک بار با دو پروسس بنویسید.

Input: 0101 1110 0110 0111 11

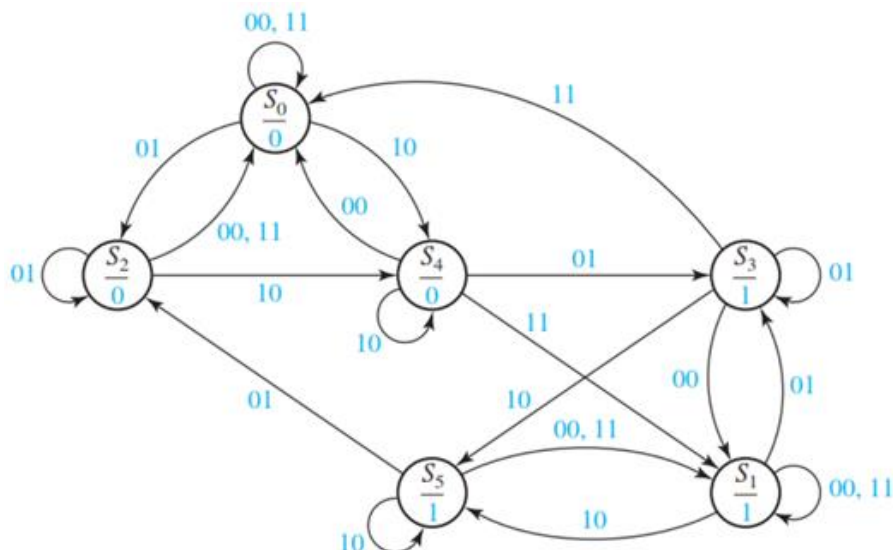
Output: 0000 0010 0100 0100 11

۷. مشکلات استفاده از ماشین Mealy در FSM را بیان کنید و بگویید چه کارهایی می‌توان انجام داد تا این مشکلات برطرف شوند.

۸. یک سیستم کنترل کننده‌ی فرآیند، ماشین حالتی به شکل زیر دارد. کد VHDL آن را با در نظر گرفتن ورودی‌ها و خروجی‌های مناسب بنویسید.

الف) این توصیف را به گونه‌ای بنویسید که خروجی‌های آن بدون تأخیر یک سیکل ساعت در ثبات قرار گیرند. برای این قسمت، کدگذاری را به صورت sequential انتخاب کنید. یک test-bench برای آن نوشته و ارائه دهید.

ب) آیا می‌توان آن را به صورت مدودف پیاده‌سازی کرد؟ اگر آری، کد توصیف مدودف آن را بنویسید.



۹. یک کنترلر ساده برای رباتی با مشخصات زیر طراحی کنید:



دانشکده مهندسی کامپیوتر

بسمه تعالی  
طراحی خودکار مدارهای دیجیتال  
نیمسال دوم ۱۳۹۴  
تمرین سوم



دانشگاه صنعتی امیرکبیر

تاریخ تحویل ۱۳۹۵/۲/۷

- روبات دارای چهار چرخ و هر چرخ به یک موتور وصل است که به صورت مستقل کنترل می شود.
- هر موتور یک ورودی تک بیتی برای کنترل سرعت دارد (یک بودن به معنی حرکت سریع و صفر بودن به معنی حرکت آهسته).
- هر موتور یک ورودی تک بیتی دیگر برای کنترل جهت حرکت دارد (یک بودن به معنی حرکت به سمت جلو و صفر بودن به سمت عقب).
- هر موتور یک ورودی تک بیتی دیگر برای روشن یا خاموش بودن دارد (یک بودن به معنی روشن و صفر بودن به خاموش).
- کاربر سه کلید برای کنترل روبات مطابق با جدول زیر دارد.

Key3	Key2	Key1	
0	0	0	Stop
0	0	1	Move forward slowly
0	1	0	Move forward fast
0	1	1	Move backward slowly
1	0	0	Move backward fast
1	0	1	Turn right
1	1	0	Turn left
1	1	1	Hold the state



۱۰. یک ALU با مشخصات زیر طراحی کنید.

entityalu is

```
port ( a : std_logic_vector (3 downto 0);  
      b :std_logic_vector (3 downto 0);  
      alucode :std_logic_vector (2 downto 0);  
      result :std_logic_vector (3 downto 0);  
      z :std_logic;  
      o :std_logic  
    );
```

endalu;

alucode	Operation
ADD	result = a+b z = carry out
SUB	result = a-b z = borrow
and	result = a AND b z = 1 if (a AND b) = 111
CMP	result = The smallest input z = 1 if a = b
RT	result = a(2 downto 0) & 0 z = a(3)
RR	result = 0 & a(3 downto 1) z = a(0)
Parity	result = a z = 1 if the number of '1's are odd in a



۱۱. برای یک Seven Segment Decoder دو پیاده سازی زیر مفروض است. با استفاده از configuration ماژولی بنویسید که با گرفتن ورودی‌های مربوط به ۴ seven segment خروجی‌های مناسب را تولید کند. دو تا از componentها از پیاده سازی اول و دوتای باقیمانده از پیاده سازی دوم استفاده می‌کنند. این کار را یک بار با configuration specification در داخل بدنه architecture و یک بار با configuration declaration در خارج از آن انجام دهید.

-----first

```
Entity sevsegv is
Port (d0, d1, d2, d3 :IN vlbit;
      a, b, c, d, e, f, g : OUT vlbit);
End sevsegv;
```

Architecture behave1 Of sevsegv IS

signal data : vlbit\_vector(3 downto 0);

begin

data<= d3&d2&d1&d0;

decode: Process(d3,d2,d1,d0)

begin

a<='1'; b<='1'; c<='1'; d<='1'; e<='1'; f<='1'; g<='1';

case data is

when "0000" => a<='0'; b<='0'; c<='0'; d<='0'; e<='0'; f<='0'; g<='1';

when "0001" => a<='1'; b<='0'; c<='0'; d<='1'; e<='1'; f<='1'; g<='1';

when "0010" => a<='0'; b<='0'; c<='1'; d<='0'; e<='0'; f<='1'; g<='0';

when "0011" => a<='0'; b<='0'; c<='0'; d<='1'; e<='1'; f<='0'; g<='1';

when "0100" => a<='1'; b<='0'; c<='0'; d<='1'; e<='1'; f<='0'; g<='0';

when "0101" => a<='0'; b<='1'; c<='0'; d<='0'; e<='1'; f<='0'; g<='0';

when "0110" => a<='0'; b<='1'; c<='0'; d<='0'; e<='0'; f<='0'; g<='0';

when "0111" => a<='0'; b<='0'; c<='0'; d<='1'; e<='1'; f<='1'; g<='1';

when "1000" => a<='0'; b<='0'; c<='0'; d<='0'; e<='0'; f<='0'; g<='0';

when "1001" => a<='1'; b<='1'; c<='1'; d<='1'; e<='1'; f<='1'; g<='0';

end case;

end process decode;

end behave1;

-----Second



### Architecture behave2 Of sevsegv IS

constanttbl : vlbit\_2d(0 to 9, 10 downto 0) := (

B"0000\_0000001,"

B"0001\_1001111,"

B"0010\_0010010,"

B"0011\_0000110,"

B"0100\_1001100,"

B"0101\_0100100,"

B"0110\_0100000,"

B"0111\_0001111,"

B"1000\_0000000,"

B"1001\_0000100"

);

signal data : vlbit\_vector(3 downto 0);

signal outv : vlbit\_vector(6 downto 0);

begin

data<= d3&d2&d1&d0;

pla\_table(data,outv , tbl);

a<=outv(0);

b<=outv(1);

c<=outv(2);

d<=outv(3);

e<=outv(4);

f<=outv(5);

g<=outv(6);

end behave2;





دانشکده مهندسی کامپیوتر

بسمه تعالی  
طراحی خودکار مدارهای دیجیتال  
نیمسال دوم ۱۳۹۴  
تمرین سوم



دانشگاه صنعتی امیرکبیر

تاریخ تحویل ۱۳۹۵/۲/۷

حل سوال ۱۲ اختیاری بوده و تحویل آن نمره اضافه خواهد شد.

۱۲. در این سوال می‌خواهیم مسابقه‌ی طناب‌کشی را شبیه‌سازی کنیم. دو بازیکن داریم و برای هر کدام یک کلید ورودی در نظر می‌گیریم. همچنین ۹ خروجی داریم (فرض کنید به LED های برد متصل می‌شوند). حالت اولیه بازی مانند شکل زیر خواهد بود که در آن خروجی وسط یک و بقیه صفر هستند:

LED1	LED2	LED3	LED4	LED5	LED6	LED7	LED8	LED9
------	------	------	------	------	------	------	------	------

در ابتدا فقط LED5 روشن است. هنگامی که بازیکن اول کلید خود را فشار دهد LED5 خاموش شده و LED6 روشن می‌شود. به همین ترتیب اگر بازیکن دوم کلید خود را فشار دهد LED فعلی خاموش و LED سمت چپ آن روشن می‌شود بنابراین هر کسی که سریعتر کلید خود را فشار دهد برنده خواهد بود. شرط برنده شدن (برای مثال بازیکن اول) این است که LED9 روشن شود و سپس بازیکن مورد نظر کلید خود را فشار دهد که در این صورت بازی تمام می‌شود و شماره‌ی برنده اعلام خواهد شد. برای مشخص کردن برنده نیز دو خروجی در نظر بگیرید که مقادیر 01 و 10 را می‌گیرد و به ترتیب برنده شدن بازیکن اول و دوم را نشان می‌دهد. همچنین یک دکمه‌ی reset در نظر بگیرید که در صورت فشار داده شدن به حالت اولیه بازی باز می‌گردد.

الف) FSM ای برای این سوال طراحی کنید. نام کدگذاری و دلیل انتخاب آن را بیان کنید.

ب) کد VHDL و test bench مربوط به قسمت الف را بنویسید.

فرض کنید مدار ما به کلاکی با فرکانس 50MHz متصل است. دقت کنید از آنجاییکه سرعت کلاک بسیار بالا است هر بار که کلیدهای ورودی فشار داده می‌شوند برای چندین دوره‌ی کلاک مقدار آن‌ها برابر یک خواهد بود بنابراین کد شما باید به گونه‌ای باشد که فقط یکبار آن را تأثیر دهد.