



تحويل در روز دوشنبه مورخ ۱۳۹۸/۰۲/۳۰ ساعت ۲۳:۵۵ از طریق سایت درس

نکاتی در رابطه با نوع تمرین

هر سری از تمرین‌ها، از چهار بخش تشکیل شده است:

- مرور و تثبیت مفاهیم: پرسش‌های این بخش جهت مرور و یادآوری مفاهیم درسی آورده شده است و با مطالعه مفاهیم درسی گفته‌شده در کلاس درس و اسلایدهای درس خواهید توانست به آن‌ها پاسخ دهید. پاسخ آن‌ها مورد ارزیابی قرار نخواهد گرفت، لذا نیازی به ارسال پاسخ آن‌ها نیست.
- تحلیل و طراحی مدار: پرسش‌های این بخش جهت درک عمیق مفاهیم درسی و افزایش قدرت تحلیل و طراحی سیستم‌های دیجیتال آورده شده است. پاسخ به آن‌ها الزامی بوده و مورد ارزیابی قرار خواهد گرفت.
- توصیف و پیاده‌سازی: پرسش‌های این بخش جهت افزایش مهارت شما در پیاده‌سازی مدارهای دیجیتال، بررسی درستی عملکرد آن و استفاده از ابزارهای طراحی آورده شده است. پاسخ به آن‌ها الزامی بوده و مورد ارزیابی قرار خواهد گرفت.
- طراحی و پیاده‌سازی سامانه پیشرفته: پرسش‌های این بخش ممکن است کمی پیچیده‌تر و دشوارتر از سایر بخش‌ها باشد. الزامی یا اختیاری بودن آن‌ها در صورت پرسش ذکر شده است.

نکاتی در رابطه با نحوه ارسال تمرین

ارسال تمرینات به صورت الکترونیکی و از طریق سایت دروس خواهد بود. فایل ارسالی شما فایل zip با نام [studentID].HW#.zip است که studentID شماره دانشجویی و HW# شماره سری تمرین است. یک قالب آماده در سامانه دروس قرار داده شده است تا پاسخ تمرین را در قالب تعیین‌شده بنویسید. پرسش‌هایی که پاسخ آن‌ها ماهیت تشریحی و تحلیلی دارد را مانند فایل نمونه در یک فایل PDF بنویسید و برای پرسش‌هایی که ماهیت کد نویسی دارند یک پوشه با نام آن ایجاد کرده و در داخل آن کدها و سایر فایل‌ها را قرار دهید.

زمان تحويل هر سری از تمرینات مشخص بوده و پاسخ تمرین پس از موعد مقرر شده در سایت درس قرار داده خواهد شد لذا امکان تغییر آن وجود ندارد. در حل تمرینات، می‌توانید به صورت دوتایی یا چندتایی با هم همفکری و بحث نمایند ولی هر شخص می‌بایست در نهایت جواب و استدلال خود را به صورت انفرادی بنویسد و در صورت شباهت پاسخ، تمامی افراد نمره تمرین را از دست خواهند داد.

چنانچه ابهامی در زمینه تمرینات دارید، می‌توانید اشکالات خود را از طریق پست الکترونیکی زیر با موضوع DA.2019 رفع نمایید.

ali.mohammadpour@aut.ac.ir

محمدپور

موفق و پیروز باشید!

تحويل در روز دوشنبه مورخ ۹۸/۰۲/۳۰ ساعت ۲۳:۵۵ از طریق سایت درس

پرسش‌های مرور و تثبیت مفاهیم

(نیازی به ارسال پاسخ این بخش نیست.)

الف) روش‌های کدگذاری حالات مختلف یک ماشین حالت را نام برده و مزایا و معایب هریک را بیان کنید.

ب) توضیح دهید در چه صورت سیگنال‌ها و متغیرهای موجود در یک فرایند^۱ در حین سنتز به حافظه تبدیل می‌شوند.

پ) فرض کنید جهت طراحی یک شمارنده که ورودی فعال‌ساز همگام و ریست غیرهمگام دارد از قطعه کد زیر استفاده شود.

- آیا ممکن است کد توصیف‌شده مشکلی ایجاد کند؟ توضیح دهید.
- در صورتی که کد مشکلی دارد آن را تغییر دهید تا یک شمارنده دارای فعال‌کننده را پیاده‌سازی کند. همچنین فضای کمتری اشغال کند و Latch ناخواسته نداشته باشد.

```
-- count is std_logic_vector(7 downto 0)

active_clock <= clock AND enable;

PROCESS (active_clock)
BEGIN
  IF reset = '1' THEN
    count <= (OTHERS => '0')
  ELSE
    IF RISING_EDGE active_clock THEN
      count <= count + '1';
    END IF;
  END IF;
END PROCESS
```

ت) مزایا و معایب طراحی با استفاده از هسته‌های مالکیت معنوی^۲ را توضیح دهید. چرا استفاده از آن‌ها ممکن است مشکلات امنیتی برای سیستم ایجاد کند؟

ث) تفاوت هسته‌های مالکیت معنوی نرم^۳، سفت^۴ و سخت^۵ در چیست. مزایا و معایب هر کدام را توضیح دهید.

^۱ Process

^۲ Intellectual Property Cores

^۳ Soft

^۴ Firm

^۵ Hard

تحويل در روز دوشنبه مورخ ۱۴۰۲/۰۲/۹ ساعت ۲۳:۵۵ از طریق سایت درس

پرسش‌های تحلیل و طراحی مدار، توصیف و پیاده‌سازی

(ارسال پاسخ این بخش الزامی است.)

(الف) مازول محاسبه‌کننده‌ی تابع $f(x) = x \times \left(\frac{1 + \sin\left(x \times \frac{\pi}{4}\right)}{2 + \cos\left(x \times \frac{\pi}{4}\right)} \right)$ را با استفاده از حافظه طراحی نمایید. ورودی مدار (x) یک بردار ۵ بیتی به صورت $2's complement$ است. خروجی مدار یک بردار ۱۶ بیتی در قالب ممیزشناور است. شکل ۴-۱ قالب نوع داده‌ی ممیزشناور را نشان می‌دهد.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S		Exponent						Fraction							

$$result = (-1)^S \times (1.Fraction) \times 2^{Exponent}$$

شکل ۱ قالب داده‌ی ممیزشناور

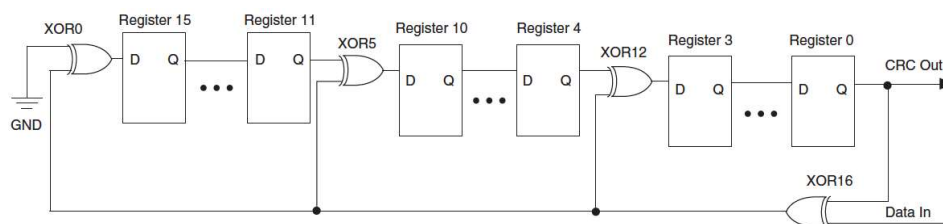
(ب) یک فایل محیط آزمون طراحی نمایید که بازای همه‌ی حالات ورودی خروجی مازول پرسش ۱ را با ساختار زیر در فایل problog1.txt بنویسد.

[Input as Vector], [Input as Integer], [Output as Vector], [Output as Real]

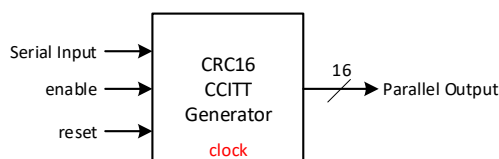
مثال

```
00000, 0, 0000000000000000, 0.0
00010, 2, 0000010000000000, 2.0
```

(۲) مازول تشخیص‌دهنده‌ی خطای CRC16-CCITT را به صورت سریالی (حساس به کلاک) طراحی نمایید. به مدار شکل زیر قابلیت Reset غیرهمگام و فعال‌ساز (Enable) همگام اضافه کنید.



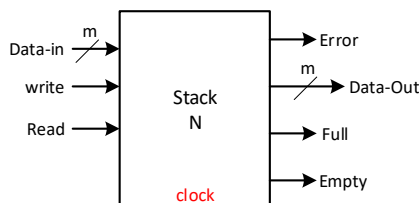
شکل ۲ مدار CCITT-CRC16



شکل ۳ CRC16 ENTITY

تحويل در روز دوشنبه مورخ ۱۳۹۸/۰۲/۳۰ ساعت ۲۳:۵۵ از طریق سایت درس

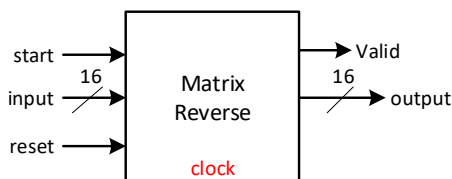
(۳) یک حافظه‌ی پشته $2^n \times m$ طراحی نمایید. 2^n تعداد خانه‌های حافظه و m طول هر یک از داده‌ها است. سیاست حافظه‌ی پشته (Last In First Out) LIFO است.



شکل ۴ STACK ENTITY

- در صورتی که Read و Write هر دو فعال باشند اولویت با نوشتن است.
- خروجی Empty زمانی فعال می‌شود که پشته خالی باشد.
- خروجی Full زمانی فعال می‌شود که پشته پر باشد.
- خروجی Error زمانی فعال می‌شود زمان سرریز^۱ یا زیرریز^۲ رخ می‌دهد.

(۴) الف) مداری طراحی کنید که وارون یک ماتریس 2×2 را محاسبه کند.



شکل ۵ MODULE ENTITY

- ماتریس به شکل $\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ است.
- مدار طراحی شده فقط یک ورودی ۱۶ بیتی برای خواندن درایه‌های ماتریس دارد. لذا لازم است این درایه‌ها به ترتیب به قطعه داده شوند. ترتیب ورود درایه‌ها به ترتیب (از چپ به راست) $a_{11}, a_{12}, a_{21}, a_{22}$ است.
- خروجی مدار باید پس از آماده‌سازی درایه‌های ماتریس وارون‌یافته را (با ترتیبی مشابه ورودی‌ها) در قالب بردار خروجی در هر کلاک ارسال کند. هنگامی که داده‌های خروجی معتبر هستند، خروجی Valid باید فعال (یک) شود.
- اعداد ورودی و خروجی از نوع ممیز ثابت $Qm.n$ از نوع Q6.9 است.

$$-2^m + \sum_{i=0}^{m-1} 2^i + \sum_{j=1}^n 2^{-j}$$

ب) یک فایل محیط آزمون طراحی نمایید که درستی مدار طراحی شده را بسنجد.

^۱ Overflow

^۲ Underflow