



دانشگاه صنعتی شریف

دانشکده مهندسی برق

درس مدارهای منطقی و سیستم دیجیتال و آزمایشگاه (۲۵۷۴۳)

## آزمایش شماره ۵

تهیه کنندگان:

دکتر محمد رضا موحدین

مجتبی پورعلی محمدی

حسین چیت ساز

## مقدمه:

تا قبل از این آزمایش، شما از مدارهای ترکیبی استفاده می کردید. مدار ترکیبی مداری است که خروجی ها تنها وابسته به ورودی های مدار است و اثری از المان حافظه در مدار وجود ندارد. در این آزمایش شما با مدارات ترتیبی آشنا می شوید. مدار ترتیبی مداری است که خروجی ها علاوه بر ورودی ها، به استیت مدار (مقادیری که المان های حافظه ذخیره کرده اند) که می تواند اثری از ورودی های قبلی مدار باشد نیز بستگی دارد. مدارهای ترتیبی به ما اجازه ی استفاده از زمان در مداراتمان را می دهند.

در این دستورکار به صورت پله پله انواع المان های حافظه که در مدارات دیجیتال وجود دارند مانند لچ ها و فلیپ فلاپ های مختلف را با استفاده از گیت های ترکیبی ساخته و بر روی FPGA پیاده سازی می کنیم و در بخش بعدی از رجیسترهای موجود در FPGA برای این کار کمک میگیریم.

همانطور که میدانید، به هیچ وجه نباید port های تاپ ماجولی که به شما داده شده است را تغییر دهید، لذا برای جلوگیری از پیچیدگی نیاز است در هر قسمت از دستور کار ماجولی با نام مناسب (مانند gated\_D\_latch) درست کنید و آن را صرفاً در ماجول instantiate, top\_module\_extended کنید و ورودی ها و خروجی های آن را به کلید ها یا led های خواسته شده متصل کنید.

پس از انجام دادن هر قسمت، RTL schematic را در بخش elaborate نرم افزار Vivado چک کنید. در تمام بخش های این دستور کار، علاوه بر اینکه مدارتان با استفاده از بورد توسط دستیار چک می شود، نیاز است کد ورپلاگ و RTL schematic را نیز به دستیار نشان دهید.

برای قسمت هایی که مدار دو ورودی دارد از button\_1 و button\_2 استفاده کنید و برای قسمت هایی که تعداد ورودی بیشتری دارند از button\_1 و button\_2 و button\_mb استفاده کنید (فراموش نکنید که button\_mb ها active low هستند). خروجی ها را به led[0] و led[1] متصل کنید.

## بخش اول:

در هر کدام از قسمت های این بخش، یک المان حافظه با استفاده از گیت های درونی وریلاگ توصیف میکنیم.

برای ساختن المان حافظه نیاز به ایجاد combinational loop است که بتواند یک state پایدار را درون خود ذخیره کند. حلقه ی ترکیبی حلقه ای در مدار است که با شروع از هر کدام از سیم های آن حلقه و رد شدن از گیت های ترکیبی می توانیم به همان سیم برگردیم. در واقعیت ایجاد حلقه ی ترکیبی کار غلطی است و در صورت اینکه Vivado آن را تشخیص دهد، error می دهد و به شما اجازه ی پیاده سازی روی برد را نمی دهد. برای جلوگیری از این ارور، این دستور را در خط آخر فایل constraint اضافه کنید:

```
set_property ALLOW_COMBINATORIAL_LOOPS TRUE [get_nets *];
```

دقت کنید که فقط و فقط در این بخش از این دستور کار به دلیل مسائل آموزشی شما مجاز به ایجاد لوپ ترکیبی در مدارتان هستید. هدف آشنا شدن شما با مدار داخلی و نحوه ی کار کردن المان های حافظه مانند لچ و فلیپ فلاپ است. در بخش بعدی و دستور کار های بعدی از فلیپ فلاپ های موجود در خود FPGA (که خودشان با استفاده از حلقه ی ترکیبی درست شده اند) استفاده می کنیم.

در کدهای وریلاگی که برای بخش اول می نویسید، مجاز به استفاده از عبارات continuous assignment و یا always block نیستید و کد شما باید صرفاً از instance گرفتن المان های پایه ی تعریف شده در وریلاگ مانند not, and, nand و ساخته شده باشد و به عبارتی دیگر به شکل structural نوشته شده باشد.

هر کدام از ۶ قسمت زیر را توسط گیت های درونی وریلاگ توصیف کرده و بر روی بورد پروگرم کنید (عکس ها در صفحات ۲۵۰ الی ۲۶۰ نسخه ی سوم کتاب Brown هستند که در CW درس نیز آپلود شده است):

- a) A basic latch built with NOR gates (Figure 5.4 a)
- b) Gated SR latch (Figure 5.5 a)
- c) Gated SR latch with NAND gates (Figure 5.6)
- d) Gated D latch (Figure 5.7 a)
- e) Master-slave D flip-flop (Figure 5.9 a)

در قسمت e، از Gated D latch ی که در قسمت d درست کردید instance بگیرید.

- f) A positive-edge-triggered D flip-flop (Figure 5.11 a)

## بخش دوم:

در این بخش کد وریلاگ را با استفاده از `always block` طوری مینویسیم که از خود المان های حافظه ی موجود در `FPGA` استفاده شود. در عمل نیز همواره از این روش استفاده خواهید کرد و صرفا به دلایل آموزشی روش های بخش قبل را انجام دادیم. دقت کنید که `coding style` شما در این بخش بسیار مهم است.

ابتدا دستوری را که در خط آخر فایل `constraint` اضافه کردید را حذف کنید سپس هر کدام از ۷ قسمت زیر را توسط توصیف `behavioral` پیاده سازی کرده و بر روی بورد پروگرم کنید (عکس های سه قسمت اول در صفحات ۲۵۰ الی ۲۶۰ نسخه ی سوم کتاب `Brown` هستند که در `CW` درس نیز آپلود شده است. برای چهار قسمت بعدی می توانید به اسلاید های درس مراجعه کنید):

- a) D-Latch (Figure 5.7 c)
- b) negative edge D-FF (Figure 5.9 c)
- c) positive edge D-FF (Figure 5.11 b)
- d) D-FF with async. preset and clear
- e) D-FF with sync. preset and clear
- f) T-FF
- g) JK-FF