



به نام خدا



طراحی کامپیوتری سیستم‌های دیجیتال - پاییز ۱۴۰۳

پروژه 3: سنتز ضرب کننده‌ی تقریبی با استفاده از

ماژول های Actel

طراحان: [هاتف رضائی](#) - [شهزاد ممیز](#)

هدف پروژه :

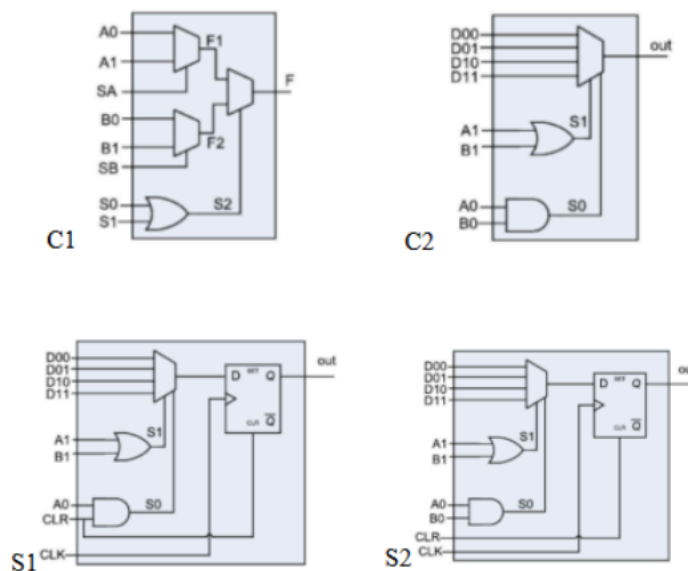
در این تمرین می خواهیم با نحوه ی سنتز شدن کد ورپلاگ روی FPGA آشنا شویم به این منظور؛ باید مسئله ی ساده تر شده ی تمرین یک را به صورت دستی توسط ماژول های Actel پیاده سازی کنیم.

صورت مسئله:

مسئله ی این تمرین ساده شده ی تمرین اول (ضرب کننده تقریبی) است. در این تمرین برای ساده تر شدن مسئله دو عدد 16 بیتی به عنوان ورودی به ماژول شما داده خواهند شد. از سمت چپ اعداد، صفر ها را دور میریزیم تا به اولین 1 برسیم و به اندازه ی 8 بیت از آنها جدا کرده و این دو عدد 8 بیتی را در هم ضرب می کنیم. سپس نتیجه را به اندازه ی صفر هایی که از سمت چپ اعداد ورودی دور ریختیم به سمت راست شیفت داده و به عنوان خروجی ماژول در نظر میگیریم. (خروجی 16 بیتی است).

آشنایی با ماژول های Actel:

ماژول های منطقی Actel که در شکل 1 نشان داده شده‌اند ماژول های منطقی قابل برنامه‌ریزی هستند. با استفاده از آن‌ها می توان مدار های combinational و sequential های مختلفی را پیاده سازی کرد. هدف این تمرین، پیاده سازی و سنتز طراحی بر روی بخشهای قابل برنامه ریزی یک FPGA هستند. بنابراین ضروری است تا مسیرهای داده و کنترل را ابتدا به صورت ساختاری یا همان Structural طراحی کنید و سپس طرح را بر روی سلول های منطقی قابل برنامه ریزی معرفی شده، سنتز کنید. سلولهای منطقی در دسترس در شکل ۱- نمایش داده شده‌اند. دقت کنید که برای طراحی مسیر داده و کنترلر (FSM) شما تنها مجاز به استفاده از سلول های s1، s2، c1، c2 هستید.



شکل 1. سلول های قابل برنامه ریزی مربوط به ماژول Actel

برای این تمرین ابتدا کد مربوط به ماژول هایی با نام های S2، S1، C2، C1 را بنویسید که توصیف ماژول های ارائه شده هستند. سپس بقیه ماژول های لازم را که در پروژه ی یک طراحی کردید با اتصال این ماژول ها پیاده سازی کنید. لازم به ذکر است که مجاز به استفاده از عملگر های منطقی در هیچ یک از ماژول ها به جز S1، C2، C1 S2 نیستید و فقط مجاز به استفاده از ماژول های شکل 1 هستید.

مواردی که در حین پیاده سازی باید در نظر بگیرید:

نحوه گرفتن ورودیها و خروجی دادن برنامه، باید دقیقا مطابق با پروژه اول درس باشد. اکیدا توصیه می شود طراحی به صورت سلسله مراتبی انجام شود. یعنی از ماژول های S2، S1، C2، C1 استفاده و با آنها سایر ماژول ها را به صورت سلسله مراتبی پیاده سازی و استفاده کنید. به صورت آزاد از ماژول ها استفاده نکنید! اگر تعداد گیت هایی (جمع گیت ها به ازای هر ماژول) که برای پیاده سازی استفاده کرده اید از 12000 تا بیشتر شوند به نسبتی که گیت اضافه استفاده کرده اید از نمره ی شما کسر خواهد شد.

تا جای ممکن کنترلر خود را ساده تعریف کنید تا در ساده کردن جدول کارنوی مربوطه و سپس پیاده سازی آن با ماژول های تعریف شده به مشکل نخورید. (می توانید از one hot استفاده کنید)

مواردی که باید تحویل دهید:

- طراحی کنترلر و مسیر داده به صورت دستی (برای همه ی ماژول ها)
- کدهای مربوط به زبان ورپلاگ (تمام ماژول ها)
- فایل da.txt

امتیازی:

- اگر ماژول های استفاده شده از 6000 تا کمتر باشد 5% نمره ی امتیازی می گیرید.
- به سه گروهی که کمترین تعداد گیت را دارند (به شرط آنکه از 6000 تا کمتر گیت استفاده کرده باشند و عملکرد ماحول کاملاً درست باشد) به ترتیب 10، 15، 20 نمره ی امتیازی تعلق می گیرد.

روش شمردن تعداد ماژول ها:

کد های **cpp** داده شده را کامپایل کرده و فایل اجرایی (.exe) را در کنار فایل های پروژه بگذارید . لازم به ذکر است اسم فایل های اجرایی باید هم اسم ماژول ها باشد تا دستور \$system آن را اجرا کند (اگر اسم فایل خروجی را تغییر دادید باید در دستور سیستم هم تغییر ایجاد کنید). قبل از اجرای simulation فایل da.txt را نیز در کنار پروژه قرار داده و در آن یک 0 بنویسید تا با اجرا شدن برنامه عدد آن به ازای هر instantiate درون فایل به مقدار تعداد گیت های آن زیاد شود و در پایان simulation تعداد گیت های شما درون فایل نوشته شود (زمان سیمولیشن کمی طولانی تر میشود).
تعداد گیت های هر ماژول:

S2 = 13

S1 = 13

C2 = 11

C1 = 10

دستور کامپایل کد های cpp:

```
g++ c1.cpp -o c1.exe
```

```
g++ c2.cpp -o c2.exe
```

```
g++ s1.cpp -o s1.exe
```

```
g++ s2.cpp -o s2.exe
```

سایر نکات

- - انجام این تمرین به صورت گروه های دونفره خواهد بود:
- برای تمرین لازم است فایل های `Testbench` و `HDL` خود را مطابق توضیح داده شده در `trunk` در `subdirectory` های `trunk/doc` بارگذاری کنید. همچنین اطمینان حاصل کنید که با اجرای `trunk/sim/sim_top.tcl` تست پنج شما اجرا می شود. برای اجرای این اسکریپت می توانید از دستور زیر در Modelsim استفاده کنید:

>> do <sim_file>

- فایل ها و گزارش خود را تا قبل از موعد تحویل، با نام CAD_HW3_<SID>.zip در محل مربوطه در صفحه درس آپلود کنید.
- در صورت هرگونه ابهام می توانید از طریق ایمیل یا [تلگرام](#) با طراحان در ارتباط باشید.
- نام گذاری صحیح متغیرها، تمیزی کد و توضیحات و پارامتری بودن ورودی‌های ماژول‌ها می تواند تا حدودی کاستی‌های کد را در بخش‌های دیگر جبران کند.
- - هدف این تمرین یادگیری شماسست! در صورت کشف تقلب، مطابق با قوانین درس برخورد خواهد شد.

موفق باشید