



## طراحی کامپیوتری سیستم های دیجیتال

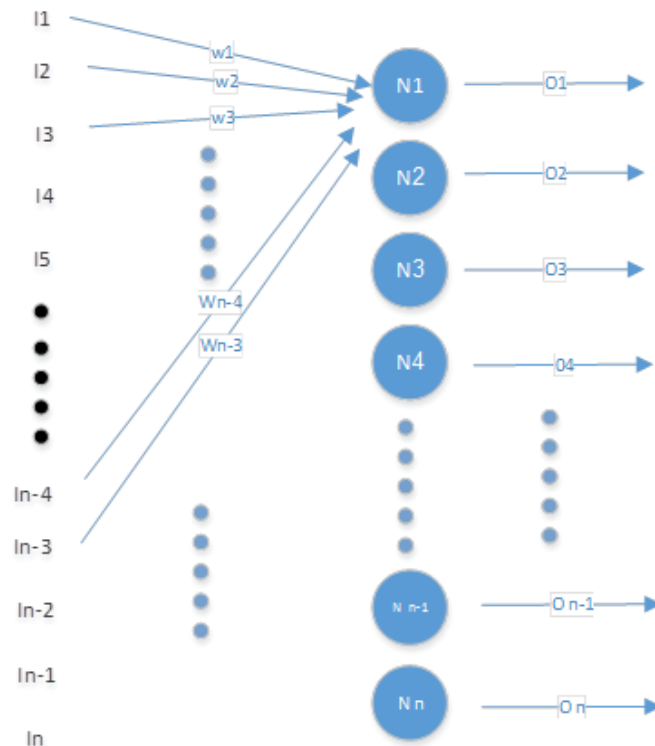
(سال تحصیلی ۹۸-۹۷، نیمسال اول)

### تمرین هشتم: سنتز مدارهای ASIC و شبیه سازی بعد از سنتز

#### ۱-۱- پیاده سازی شبکه عصبی خود سازمان ده

شبکه های عصبی (Neural Networks)، سیستم های پردازشی هستند که با الهام از ساختار مغز موجودات زنده، یک چارچوب کلی برای پیاده سازی الگوریتم های مبتنی بر یادگیری را فراهم می نمایند. الگوریتم های مبتنی بر یادگیری، دسته ای از الگوریتم ها هستند که مراحل انجام آن ها به دو بخش آموزش (train) و سنجش (test) تقسیم می شود. در بخش آموزش با اعمال یک سری از ورودی ها، پارامترهای یادگیری سیستم آموزش داده می شود و در بخش سنجش با اعمال یک دسته دیگر از ورودی ها، الگوریتم، مورد سنجش قرار می گیرد. در این تمرین شما باید نسبت به مدل سازی یک شبکه عصبی خود سازمان ده (SOM (Self Organization Map که از تعدادی (MAC (Multiply Accumulator) به عنوان نرون های شبکه تشکیل شده است، اقدام نمایید.

ساختار شبکه مورد نظر در شکل زیر نشان داده شده است. هر نرون از شبکه عصبی از لحاظ سخت افزاری مانند یک MAC عمل می نماید، به طوریکه حاصل ضرب مقادیر ورودی در وزن متناظر را محاسبه می کند و مجموع همه آنها را به عنوان خروجی تولید می نماید. به عبارت دیگر خروجی هر نرون مجموع وزن دار ورودی ها است (وزن های شبکه در طول فرایند آموزش تعیین می شوند که در این تمرین مورد نظر نیست).



شبکه عصبی مورد نظر دارای ۴ نرون و ۱۰ ورودی است، که هر نرون با یک MAC که نتیجه جمع حاصلضرب وزن‌ها در ورودی‌های متناظر را محاسبه می‌کند، پیاده سازی می‌شود. مدار مورد نظر برای پیاده سازی شبکه عصبی ۳ بخش کلی دارد:

۱- ۴ عدد MAC برای تحقق ۴ نرون خروجی شبکه

۲- تعدادی رجیستر ۸ بیتی به منظور ذخیره وزن‌ها

۳- واحد مقایسه

مدار فوق باید در هر کلاک یکی از ورودی‌ها را بخواند و هر یک از چهار MAC آن ورودی را در وزن مناسب ضرب کرده و با مقادیر قبلی خود جمع نمایند. پس از اتمام خواندن ۱۰ ورودی شبکه، واحد مقایسه نسبت به مقایسه خروجی نرون‌ها و اعلام شماره نرون با بیشترین خروجی به عنوان برنده اقدام می‌نماید.

وزن‌ها و ورودی‌های شبکه فوق به صورت اعداد HEX در جدول زیر قابل مشاهده می‌باشند.

weights	1	2	3	4	5	6	7	8	9	10
Neuron1	F3	E2	FF	91	47	FF	91	47	72	61
Neuron2	12	61	32	11	A1	F3	E2	FF	91	47
Neuron3	91	47	72	61	32	11	A1	F3	78	54
Neuron4	F3	82	78	33	A1	F3	E2	FF	91	47
inputs	1	2	3	4	5	6	7	8	9	10
	12	22	63	78	91	A2	E1	D8	F5	23

مدار نهایی ۳ ورودی RST (برای صفر کردن مقدار خروجی MACها)، CLK و EN (برای شروع محاسبات) و همچنین دو خروجی ۴ بیتی و ۱۶ بیتی برای شماره نرون برنده و مقدار نهایی آن خواهد داشت.

## ۲-۱- شبیه‌سازی

پس از طراحی و نوشتن کد، طرح را با استفاده از ابزار ModelSim و ورودی‌های داده شده شبیه‌سازی نمایید. در این مرحله لازم است یک testbench مناسب تهیه گردد.

## ۳-۱- سنتز مدارهای ASIC و شبیه‌سازی بعد از سنتز

طرح پیاده‌سازی شده در قسمت قبل را با استفاده از ابزار Design Vision سنتز نمایید. به منظور استفاده از این ابزار در ترمینال سیستم عامل CentOS، دستور زیر را وارد کنید.

➤ `design_vision &`

با این دستور محیط گرافیکی ابزار باز خواهد شد. حال می‌توان با استفاده از محیط گرافیکی و یا با وارد کردن دستورات زیر در قسمت console ابزار عملیات سنتز را انجام داد.

ابتدا با استفاده از دستورات زیر کتابخانه‌های مورد استفاده را مشخص نمایید.

- `set link_library /home/cad/TECH/SC_TSMC180/sc/synopsys/typical.db`
- `set target_library /home/cad/TECH/SC_TSMC180/sc/synopsys/typical.db`

سپس روال زیر را برای سنتز طرح ادامه دهید. دستور زیر کد طرح را از لحاظ syntax بررسی می‌نماید.

➤ `analyze -library WORK -format vhd {/home/icic/your location/code.vhd}`

دستور زیر کد را از لحاظ رعایت نکات سنتز و همچنین اتصالات بین واحدها بررسی می‌نماید.

➤ `elaborate MEM -architecture your_top_module -library DEFAULT`

دستور زیر پایه کلاک سیستم و فرکانس آن را مشخص می‌نماید.

➤ `create_clock -name "clk" -period 10 -waveform { 0 5 } { clock }`

در دستور فوق عبارت clk نام پایه کلاک در کد، عدد 10 پریود کلاک بر حسب نانو ثانیه و عبارت {0 5} نحوه شکل موج کلاک را مشخص می‌نماید.

دستور زیر عملیات سنتز طرح را انجام می‌دهد.

➤ `compile -exact_map`

پس از سنتز طرح می‌توانید گزارش‌هایی از قبیل گزارش زمانبندی و مساحت طرح را با استفاده از دستورات زیر مشاهده نمایید.

➤ `report_timing`

➤ `report_area`

حال که سنتز طرح به پایان رسید، خروجی این عملیات که یک فایل netlist با قالب دلخواه (verilog / vhd) می‌باشد را با استفاده از دستور زیر تولید نمایید. این فایل در واقع ترجمه طرح اولیه به طرحی است که واحدهای تشکیل دهنده آن سلول‌های استاندارد موجود در کتابخانه مورد استفاده می‌باشند.

➤ `write -format vhd -hierarchy -output netlist.vhd`

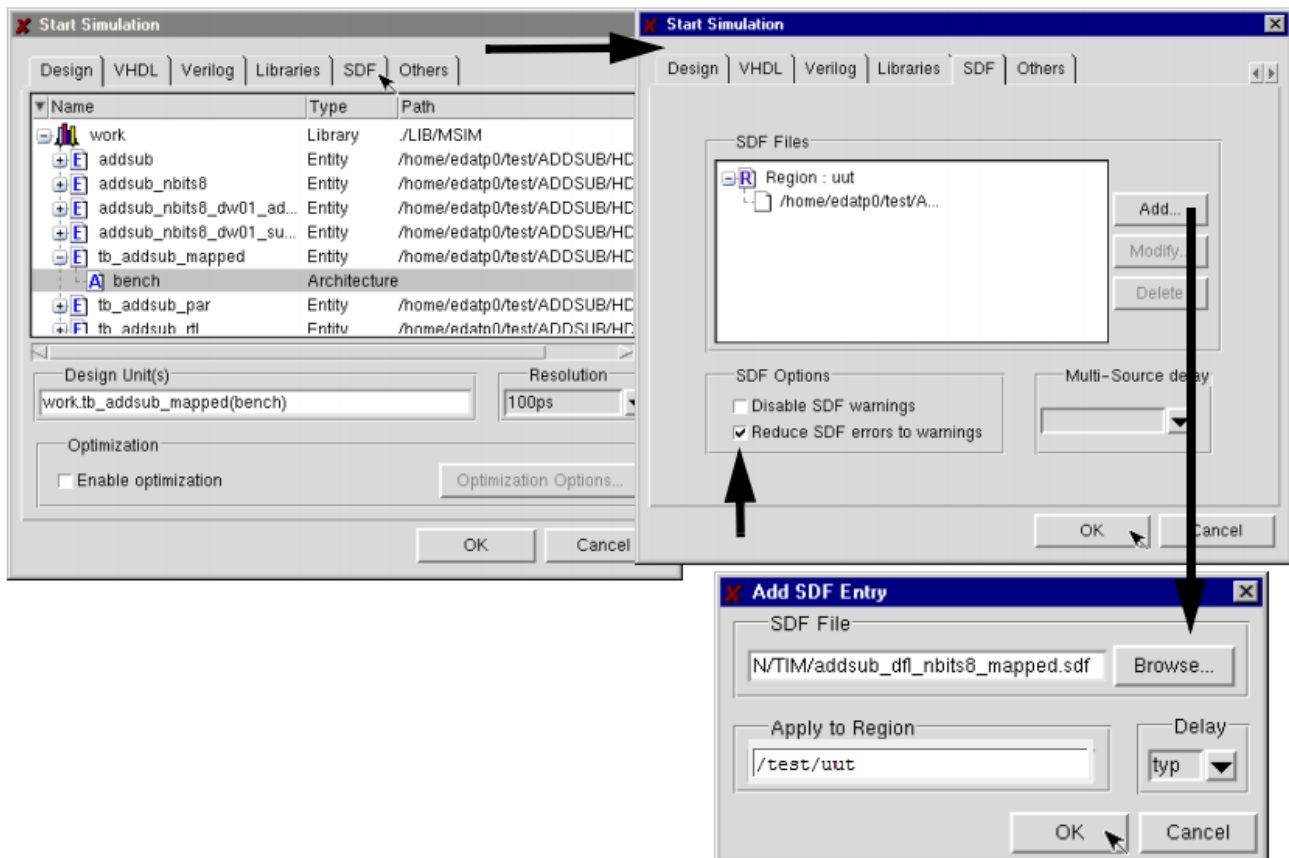
## ۴-۱- شبیه‌سازی بعد از سنتز

در این مرحله از تمرین باید netlist خروجی حاصل از مرحله سنتز را با ابزار ModelSim شبیه‌سازی نمایید. به این منظور علاوه بر فایل netlist نیاز به فایل زمانبندی سلول‌ها نیز می‌باشد. فایل زمانبندی با پسوند sdf، با استفاده از دستور زیر تولید می‌گردد.

➤ write\_sdf -version 2.1 lib\_timing.sdf

حال تمامی فایل‌های مورد نیاز برای شبیه‌سازی را از محیط سیستم‌عامل ابزار به محیط شبیه‌ساز منتقل نمایید. این فایل‌ها شامل netlist، فایل با پسوند sdf، و همچنین فایل موجود در کتابخانه مورد استفاده با پسوند vhd، می‌باشد. اکنون این فایل‌ها را به همراه کد testbench قبل از سنتز خود با استفاده از ابزار ModelSim شبیه‌سازی نمایید.

به منظور شبیه‌سازی با استفاده از فایل زمانبندی، باید در ابزار ModelSim از tab مربوط به simulation گزینه simulation start را مطابق شکل زیر انتخاب کنید. سپس فایل زمانبندی تولید شده در مرحله قبل را پیوست کرده و محل اعمال آن بر روی کد را نیز مشخص نمایید.



همانطور که در شکل فوق مشخص است برای مشخص کردن محل اعمال فایل زمانبندی در قسمت apply to region ابتدا نام واحد testbench و سپس نام مربوط به نمونه گیری (port map) از top module در کد testbench را وارد نمایید. برای مثال نحوه نام گذاری شکل فوق مطابق با کد زیر است.

```
ENTITY test IS
END test;
```

```
uut: alu port map(
```

## ۱-۵- نکات فنی:

- گزارش تمرین باید شامل گزارش زمانبندی و گزارش مساحت طرح سنتز شده باشد.
- در قسمت شکل موج‌ها یک شکل موج مربوط به شبیه‌سازی قبل از سنتز و یک شکل موج مربوط به شبیه‌سازی بعد از سنتز قرار داده شود.
- نام سیگنال‌ها دقیقاً به همان صورتی که صورت تمرین تعریف شده است، استفاده شود.