



به نام خدا

دانشگاه تهران
دانشکده فنی
دانشکده مهندسی برق و کامپیوتر

مدار های منطقی و سیستم های دیجیتال

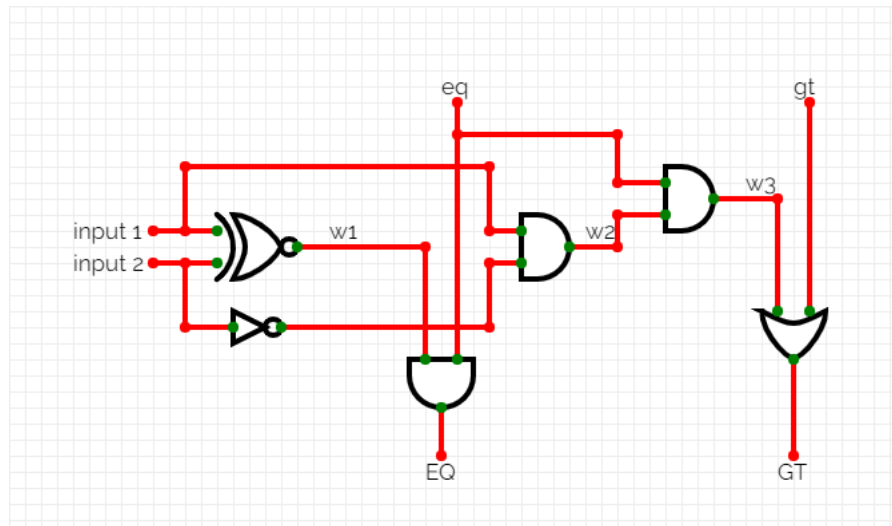
گزارش تمرین کامپیوتری سوم
CA 3

نام و نام خانوادگی:
نیلوفر مرتضوی

شماره دانشجویی:
220701096

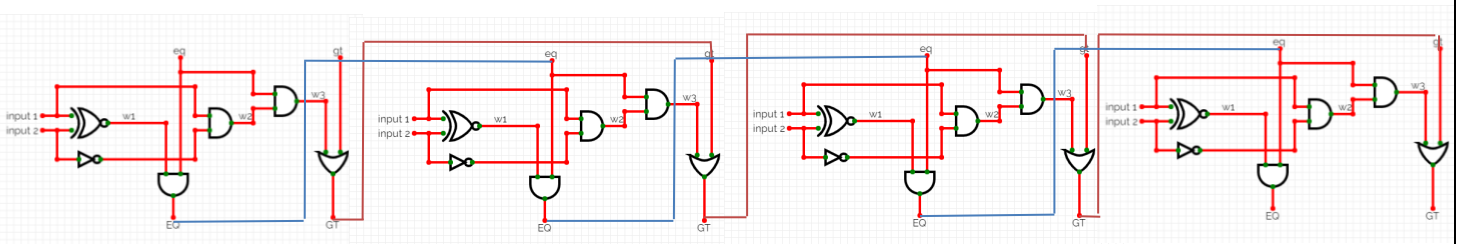
Step 1

ابتدا برای 1-bit comparator به شکل زیر gate level مدار را طراحی میکنیم و کد وریلاگ آن را مینویسیم (فایل وریلاگ 1-bit comparator):



با کنار هم گذاشتن 4 تا از این بلوک ها یک 4-bit comparator میسازیم:

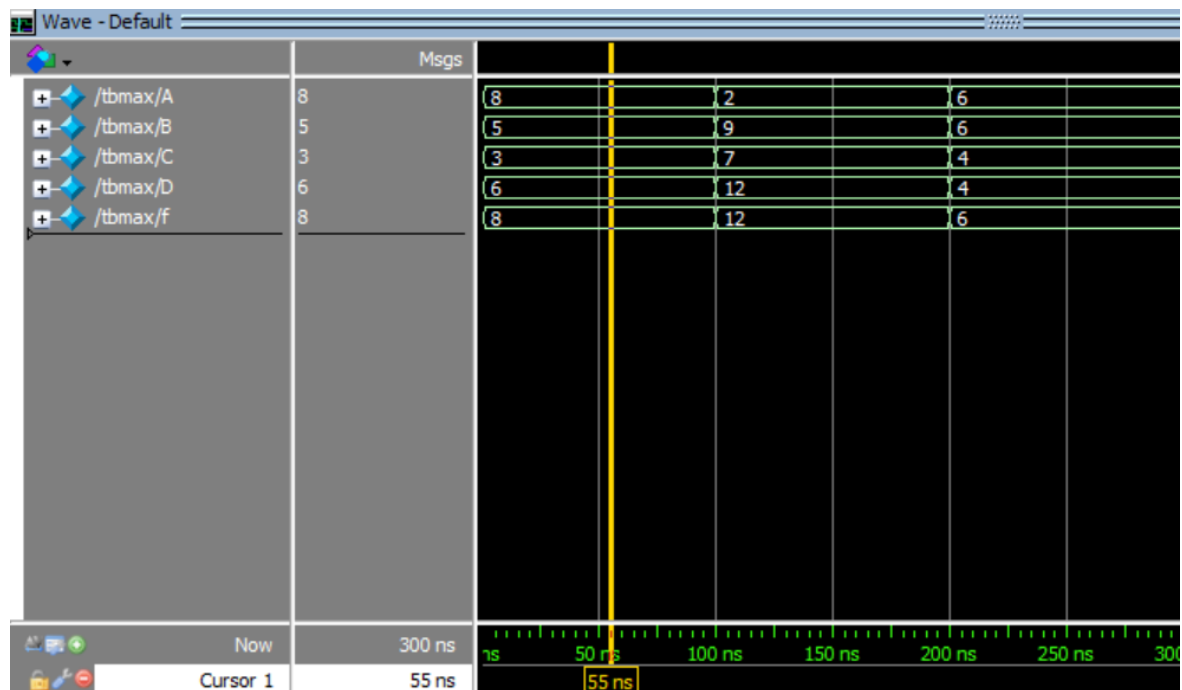
(فایل وریلاگ 4-bit comparator)



Step 2

در این گام با استفاده از 4-bit comparator ماژول یک بلاک برای تشخیص عدد بزرگتر (max-detector) مینویسیم و با یک تست بنچ صحیح بودن نحوه عملکرد آن را بررسی می کنیم:

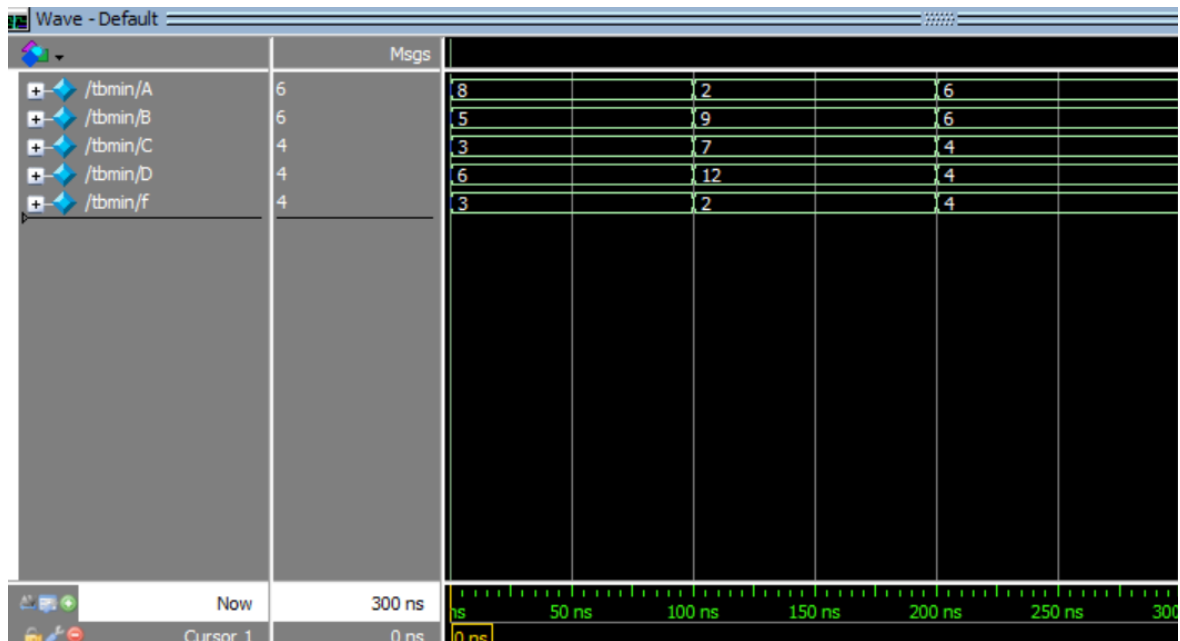
(فایل ورایلاگ max_block و maxTB)



مطابق شکل waveform بالا میتوان گفت شاهد عملکرد صحیحی هستیم.

Step 3

مانند قسمت قبل یک بلاک برای تشخیص عدد کوچکتر مینویسیم و با تست بنچ عملکرد آن را تست می کنیم. (فایل ورپلاگ minTB و min_block)

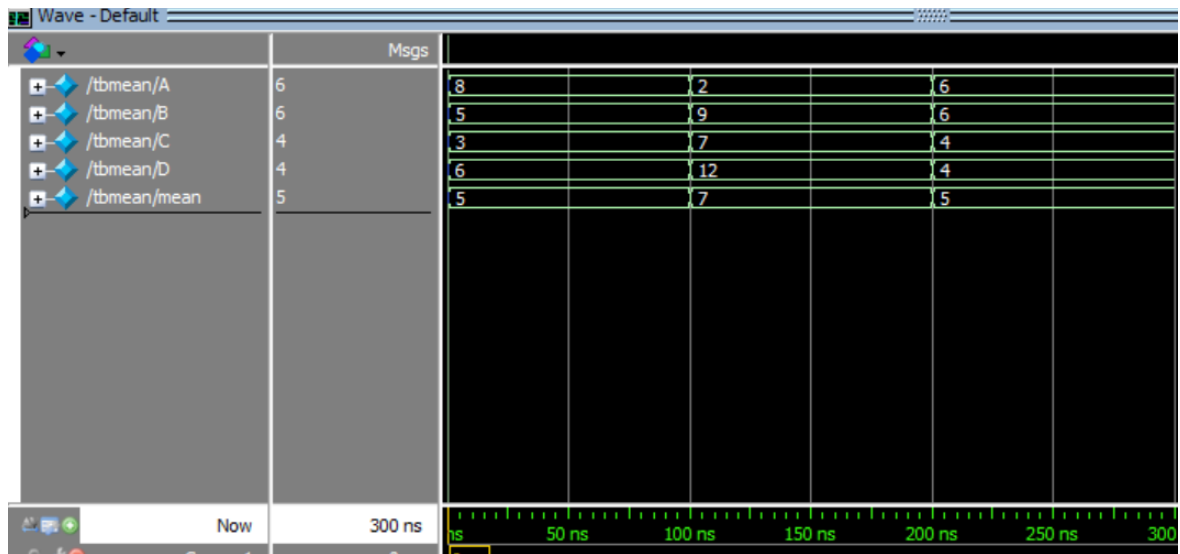


که می توان دید این بلاک هم به درستی سیگنال خروجی را تشخیص می دهد.

Step 4

ماژول mean_calculator را به صورت رفتاری مینویسیم و با تست بنچ آنرا آزمایش میکنیم:

(فایل وریلاگ mean_block و meanTB)



مشکلی که در این قسمت به آن بر میخوریم این است که اجازه استفاده از تقسیم کننده نداریم و تقسیم را به کمک right shift انجام دادیم پس مقادیرمان همیشه صحیح هستند.

مثلا در اعداد تست شده اول، میانگین درست عدد 5.5 است اما ماژول عدد 5 را به ما نشان می دهد چون از shift استفاده کردیم.

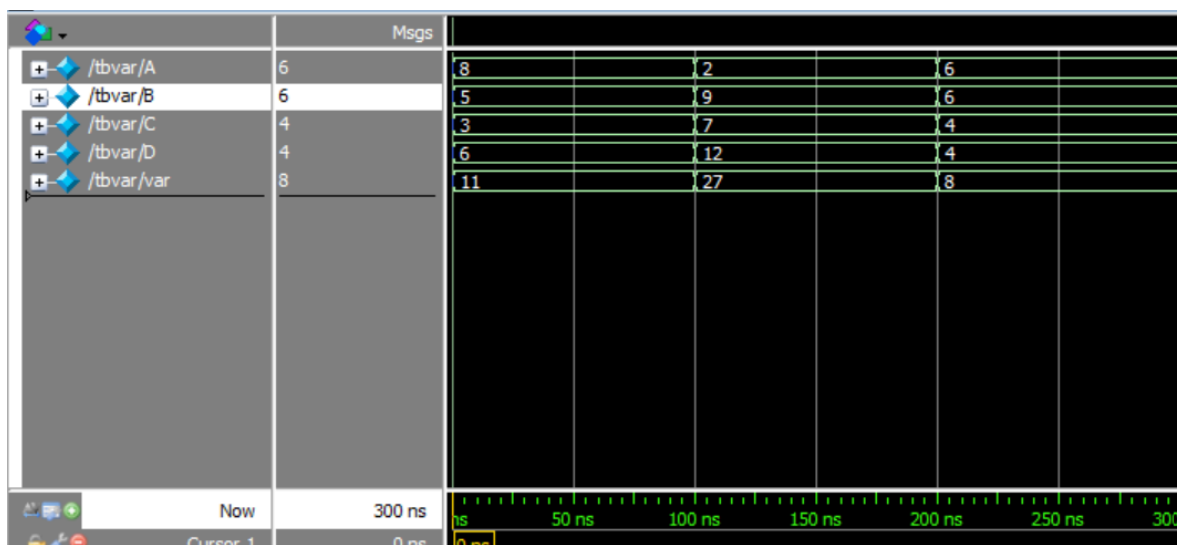
بنابراین با مقداری خطا در میانگین ها اعشاری مواجه می شویم که عدد اصلی نزدیک هستند امل کاملاً مطابقت ندارند!

Step 5

در این مرحله بصورت رفتاری ماژول var_calculator را مینویسیم که از فرمول زیر پیروی می کند:
(فایل وریلاگ var_block)

$$S^2 = \frac{\sum (x_i - \bar{x})^2}{n - 1}$$

سپس برای اطمینان از درستی کارکرد آن یک تست بنچ (فایل وریلاگ varTB) مینویسیم که waveform آن به شکل زیر است:



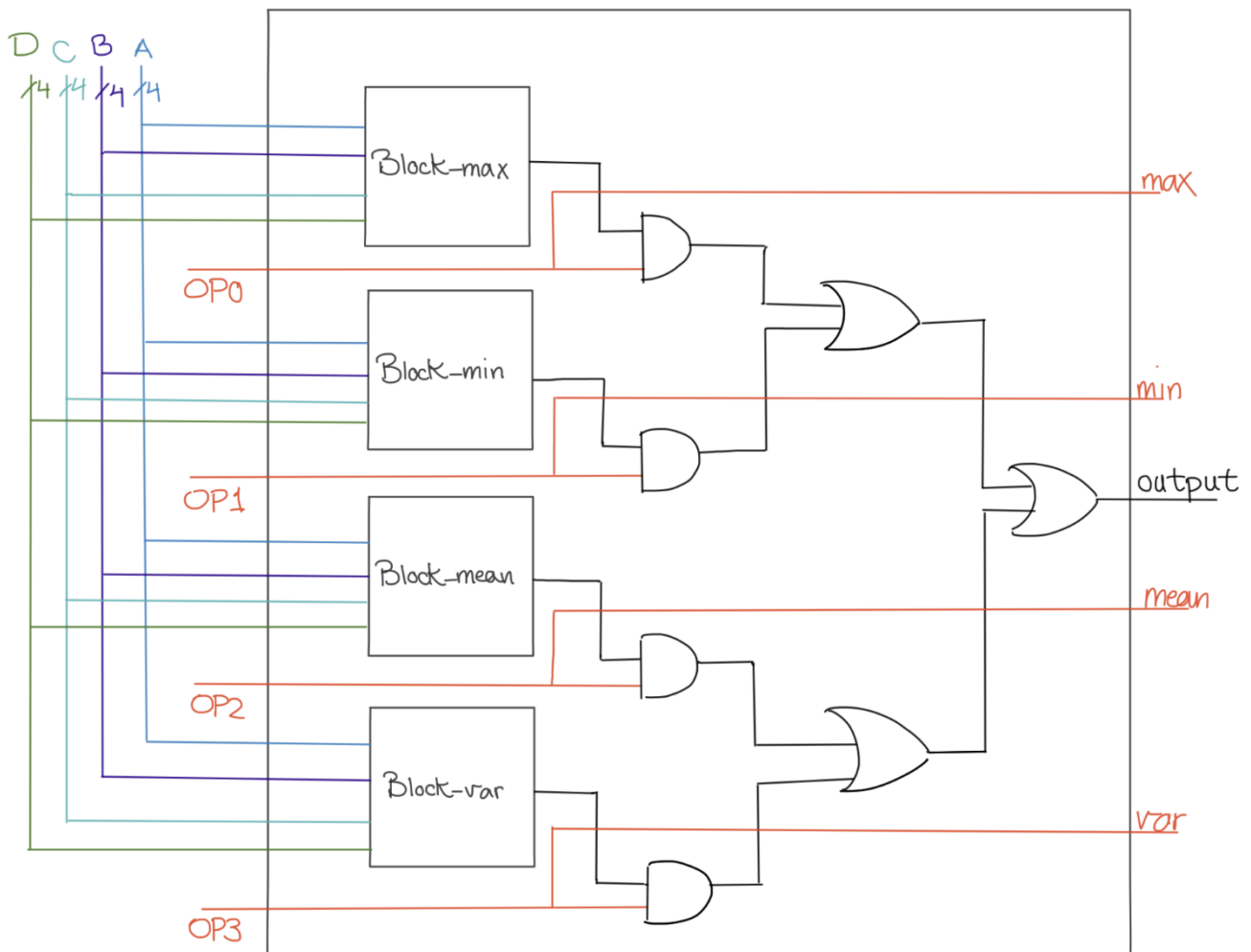
مشکلی که در این قسمت با آن مواجه هستیم مانند قسمت قبل به علت استفاده از shift به جای تقسیم کننده برای انجام تقسیمات مورد نیاز است.

بنابراین واریانس بدست آمده متفاوت از مقدار واقعی است!

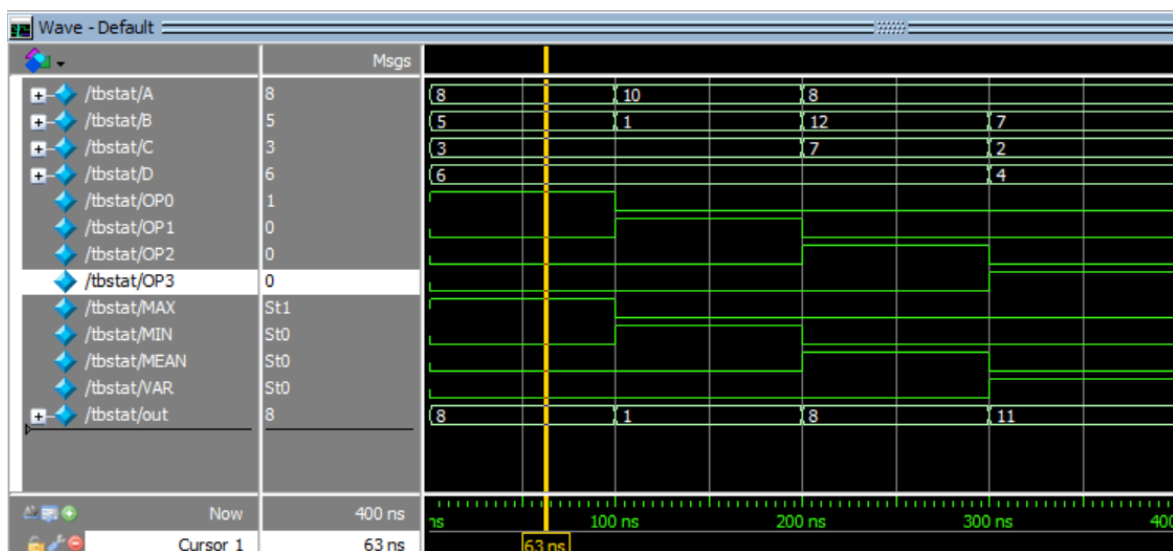
اما باقی عملکرد ماژول به درستی انجام شده.

Step 6

طراحی بلاک stat به شکل زیر است:



ماژول stat_calculator را نوشته و برای آن تست بنچ مینویسیم: (فایل ورایلاگ statTB و stat_calculator)





برای اولویت بندی سیگنال های ورودی در صورتی که بیشتر از یک سیگنال op یک باشد، هر سیگنال را با نقیض تمام سیگنال های خود and میکنیم. به صورت زیر:

assign MAX = OP0;

assign MIN = OP1 & ~OP0;

assign MEAN = OP2 & ~OP1 & ~OP0;

assign VAR = OP3 & ~OP2 & ~OP1 & ~OP0;

بدین ترتیب اولویت انجام سیگنال ها به ترتیب از 0 تا 3 است.