

گزارش کار آزمایشگاه DSD

**آزمایش شماره 4**



18 فروردین 1400

عرشیا اخوان

محمدحسین عبدی

علیرضا ایلامی

|  |  |  |
| --- | --- | --- |
| تاریخ آزمایش: 10 فروردین 1400 | موضوع: توصيف رفتاري | **شماره آزمایش: 4** |
| **علیرضا ایلامی**  **97101286** | **محمدحسین عبدی**  **97110285** | **عرشیا اخوان**  **97110422** |

هدف: طراحی یک پشته با عمق 8 و پهناي 4 بيت

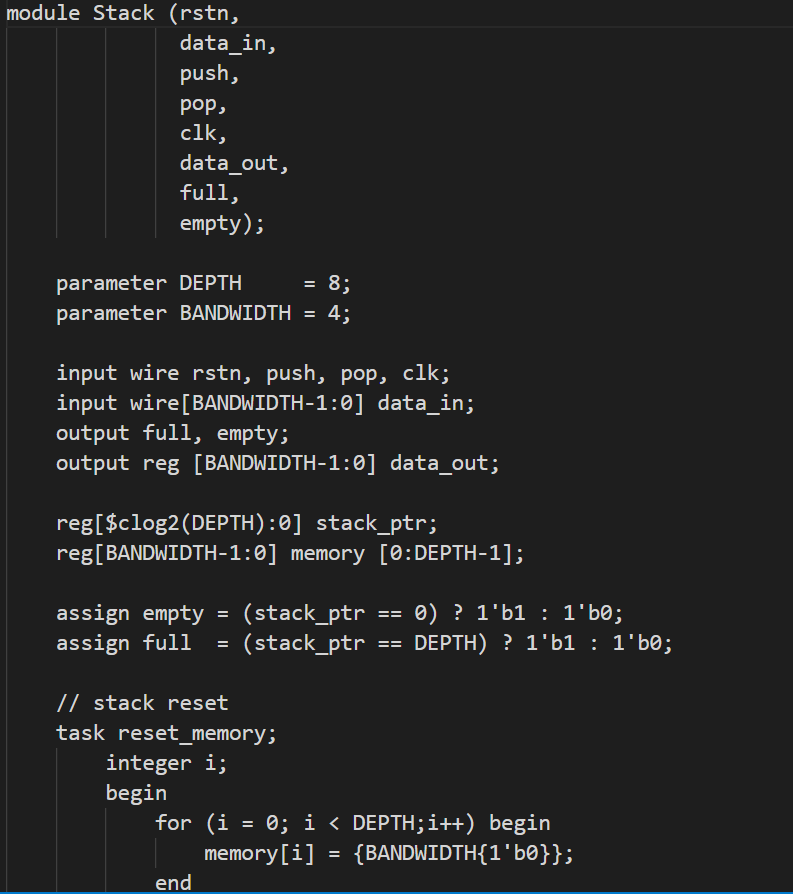
مقدمه: در این آزمایش میخواهیم یک پشته (Stack) طراحی کنیم که اعداد را یکی یکی در آن وارد کنیم. و هرگاه ظرفیت پر شد، دیگر اجازه اضافه کردن ورودی به پشته را ندهد. سپس، یکی یکی اعداد را به ترتیب FILO از پشته بیرون بیاوریم تا زمانی که به کف پشته برسیم و دیگر چیزی برای بیرون آوردن وجود نداشته باشد.

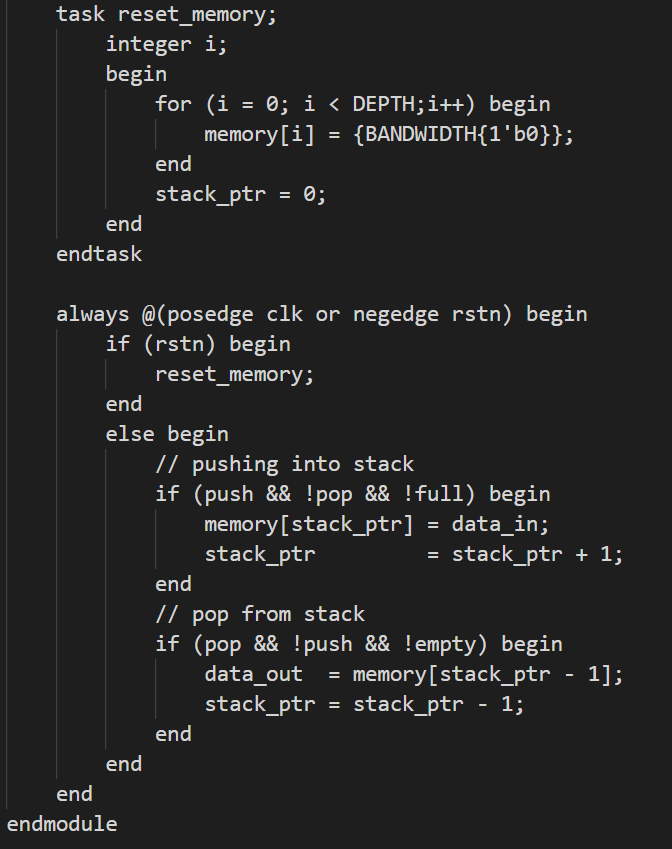
توضیح کد:

یک ماژول به نام Stack طراحی کرده ایم که تعدادی سیگنال برای push, pop و کلاک و ریست و غیره ورودی میگیرد. دو متغیر برای عمق و پهنای پشته نیز داریم که آنها را نیز ست میکنیم. این ماژول سه سیگنال خروجی دارد: یکی سیگنال data\_out (به پهنای BANDWIDTH) که عدد خروجی داده شده را در هنگام استفاده از سیگنال pop به ما میدهد، یک بیت برای نشان دادن full بودن پشته و یکی هم برای نشان دادن empty بودن پشته.

سیگنال های full , empty را به این صورت طراحی میکنیم که هرگاه نشانگر پشته یا همان stack pointer به ابتدا یا انتهای پشته اشاره میکرد، یکی از این دو بیت فعال و برابر با 1 شوند.

در ادامه، ابتدا استک را ریست میکنیم. سپس در یک always block، متناسب با سیگنالهای ورودی که کدامشان فعال باشند، اعداد را در پشته درج یا از پشته خارج میکنیم.





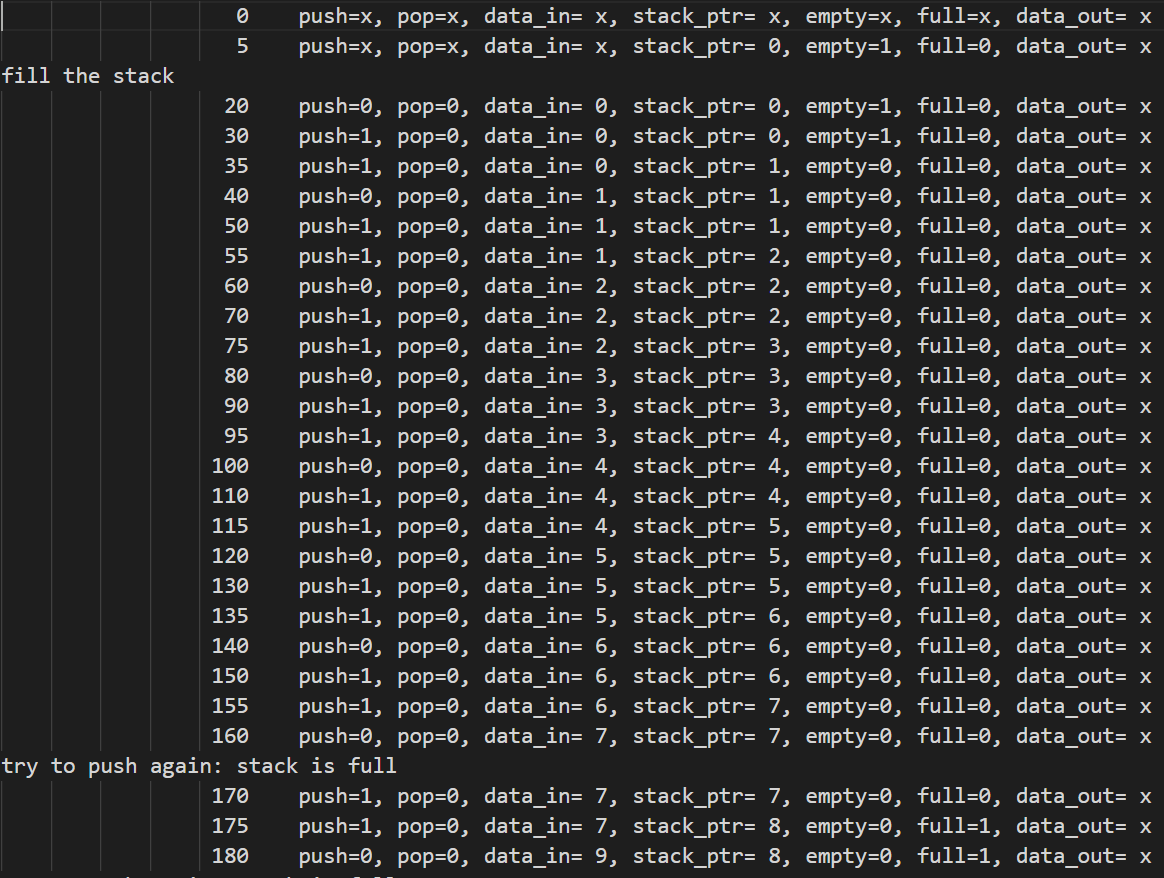
نتایج تست بنچ در پوشه report و فایل result.txt آورده شده اند.

در زیر تصویر آنها به پیوست گزارش آمده است:

توضیحات نتایج تست بنچ:

ابتدا استک ریست میشود. در ابتدا سیگنال empty را میبینیم که فعال است. سپس یکی یکی اعداد را در پشته وارد میکنیم. تا جایی که به اندازه عمق استک (در اینجا 8) عدد وارد شده باشد و استک پر شده باشد.

در اینجا اگر عدد 8 را وارد کنیم، میبینیم که چون سیگنال full = 1 است، این عدد وارد نمیشود و stack pointer تغییری نمیکند.

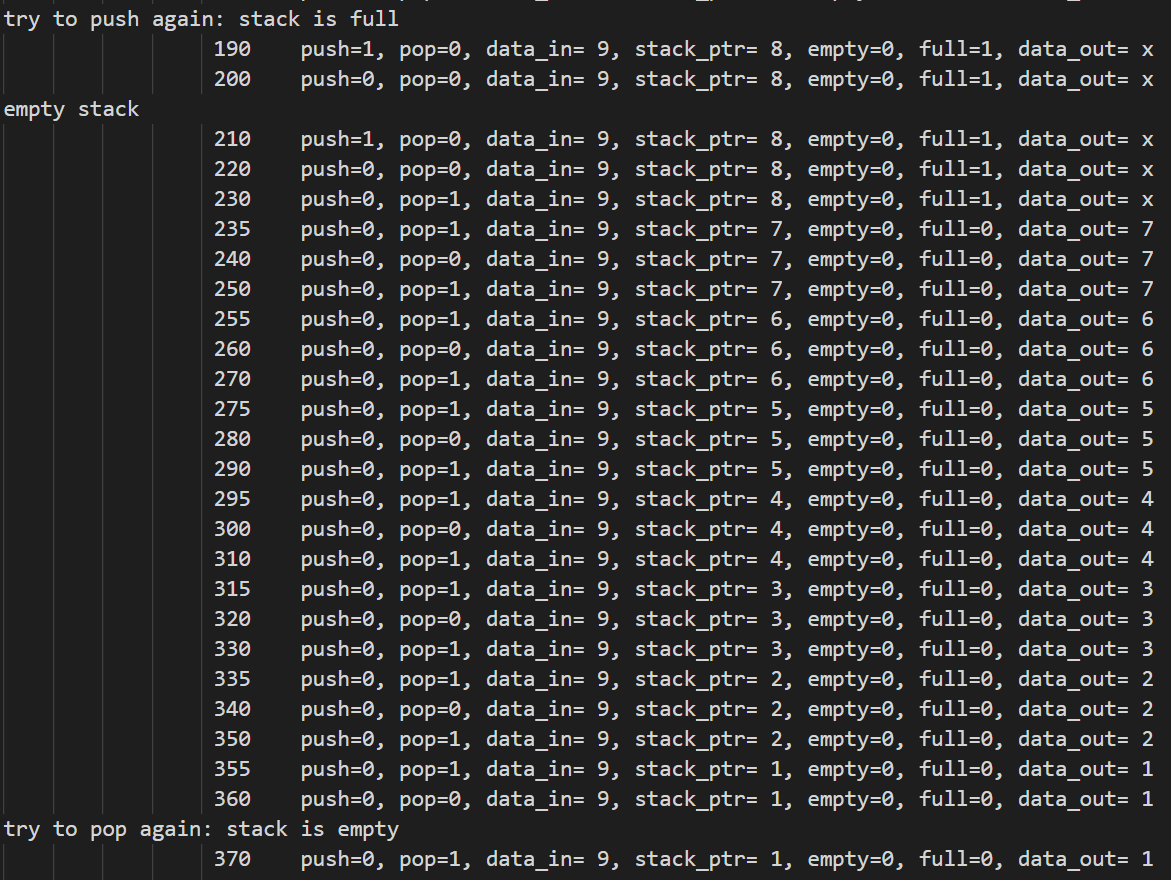


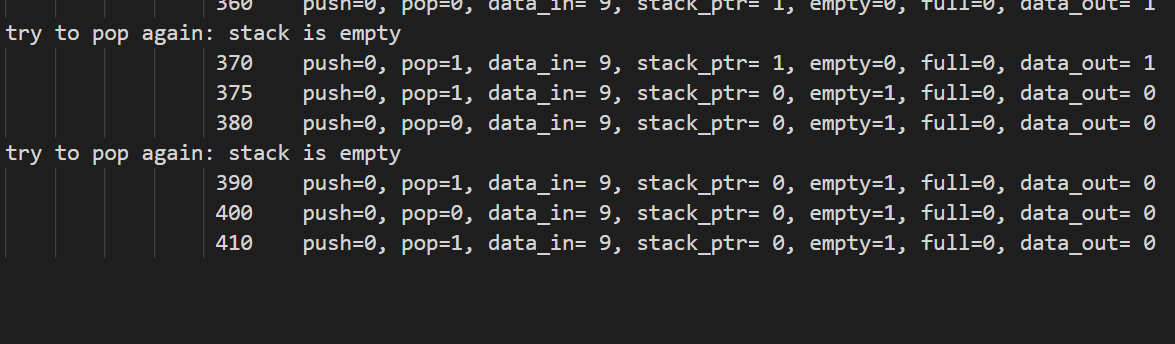
در این بخش میخواهیم استک را خالی کنیم. سیگنال pop = 1 میکنیم و سیگنال های push = 0 , empty = 0 به ما این اجازه را میدهد که از پشته عدد بیرون بیاوریم.

همانطور که واضح است. ابتدا آخرین عدد (یعنی 7) بیرون می آید.

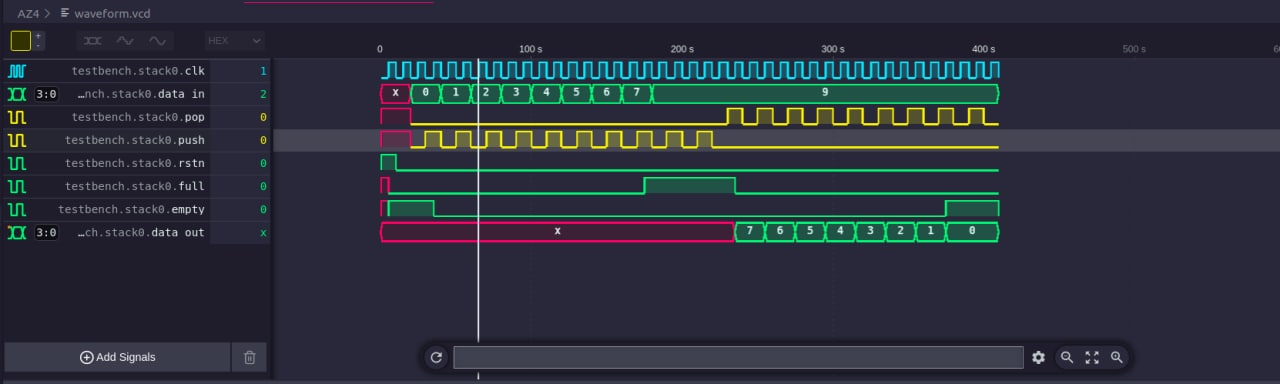
در ادامه یکی یکی بقیه خانه های استک را خالی میکنیم تا زمانی که به انتها برسیم.

هرگاه stack pointer = 0 شود به این معناست که به کف استک رسیده ایم و سیگنال empty = 1 میگردد. بدین ترتیب دیگر اجازه pop کردن نداریم و کار تمام میشود.

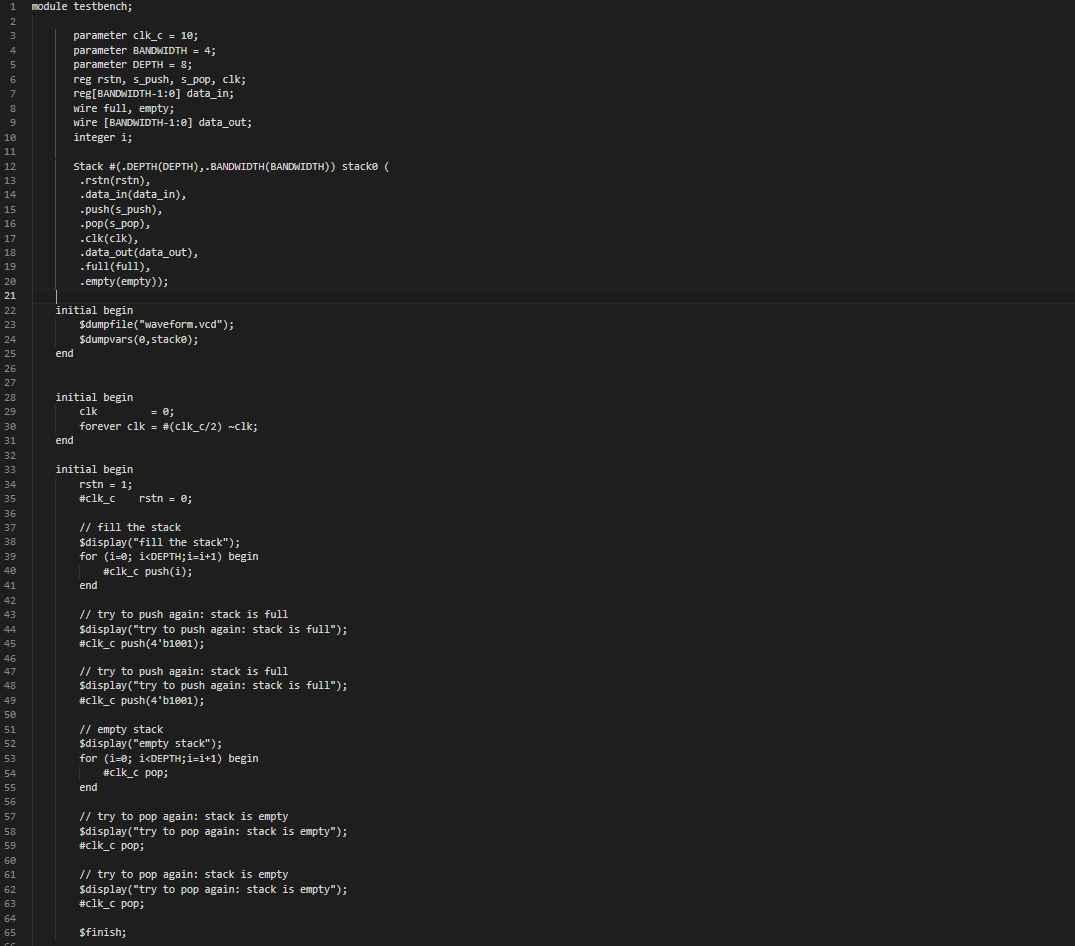


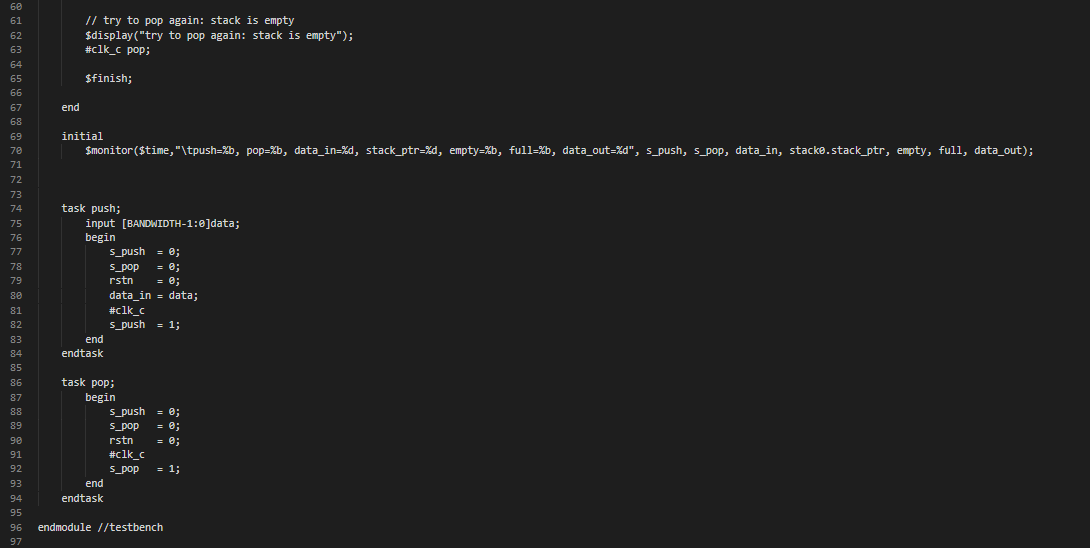


موج های خروجی به شرح زیر می باشند:



کد تست بنچ نیز در زیر آورده شده است:





***پایان***