پیش گزارش آزسیستم دیجتال 2

سهیل مهریزی 981210110

شیفت رجیستر یونیورسال :

با استفاده از شیفت رجیستر یونیورسال میتوان سه عملیات

.Parallel Load

. Shift to Right

. Shift to Left

را تنها با تغییر در ورودی سلکتور انجام داد :

Selector:

Parallel Load 00

01 Shift to Right

10 Shift to Left

شیفت رجیستر یونیورسال را میتوان با دوعنصر اساسی فلیپ فلاپ و مالتی پلکسر طراحی کرد .

در این مدار ما یک شیفت رجیستر یونیورسال 4 بیتی داریم که با مالتی پلکسر های 1-4 نوع عملیات آن تعیین میشود .

در این مدار خروجی مالتی پلکسرها به ورودی فلیپ فلاپ های نوع D متصل میشود و با سلکتور های مالتی پلکسر نوع عملیات مشخص میشود ، در خط سلکتور های 0 ورودی های بعدی فیلپ فلاپ قرار میگیرد ، در خط 1 خروجی فلیپ فلاپ قبلی (در چپ ترین عنصر ، ورودی Instance right )(Shift to right) و در خط سلکتور 2 خروجی فلیپ فلاپ بعدی (راست ترین عنصر : Instance Left)shift to left

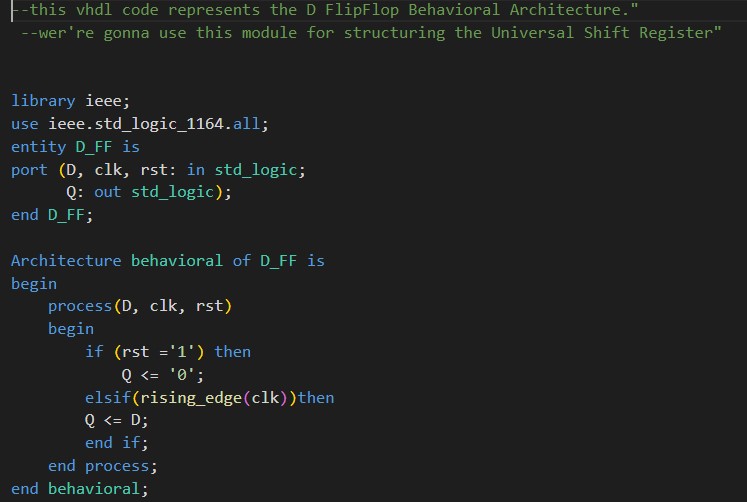
D\_Flip Flop Simulation

در این بخش از آزمایش ، شبیه سازی ماژول فلیپ فلاپ نوع Dرا داریم با استفاده ازVHDL .

شبیه سازی به دوبخش تقسیم میشود .

معماری ماژول که بصورت رفتاری طراحی شده است (BEHAVIORAL)

در بخش اول ساختار دهی رفتاری فیلپ فلاپ را طراحی میکنیم :



در ابتدای این کد کتابخانه های استاندارد IEEE را اضافه میکنیم .

در این ساختار برای فیلپ فلاپ پورت های D, CLK, RST بعنوان ورودی و پورت Q بعنوان خروجی .

در ساختار رفتاری فلیپ‌فلاپ :

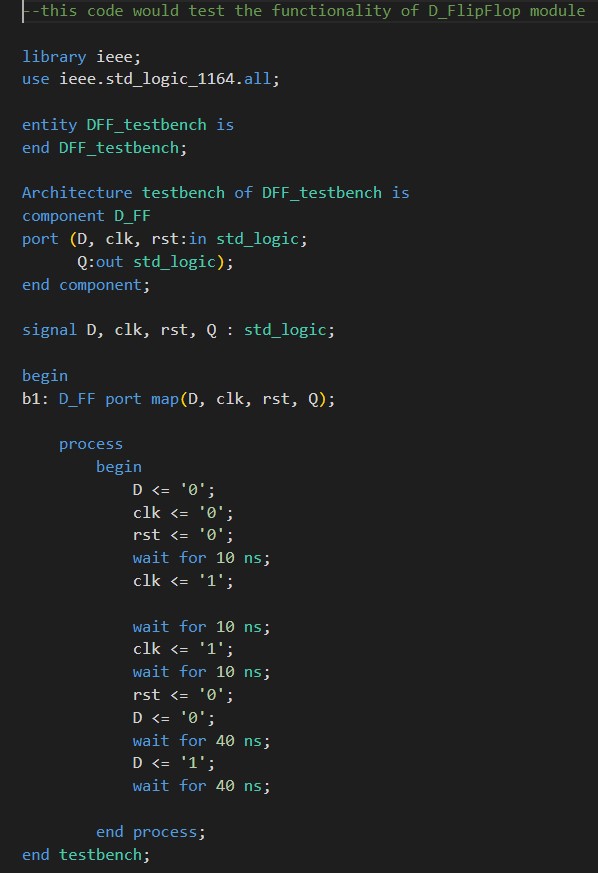
وقتی که rst برابر 1 بود ، خروجی Q را برابر صفر قرار میدهیم .

در غیراینصورت اگر rising\_edge کلاک به وقع پیوست .

ورودی D را به خروجی Q انتقال میدهد .

و در اینجا ساختار رفتاری پایان میابد ..

کد تست بنچ :



در خط کامپوننت ساختار فیلیپ فلاپ را به تست بنچ معرفی میکنیم ،

سیگنال ها را تعریف میکنیم

در پورت مپ ، در واقع سیگنال هارا به پورت های فلیپ فلاپ وصل میکنیم .

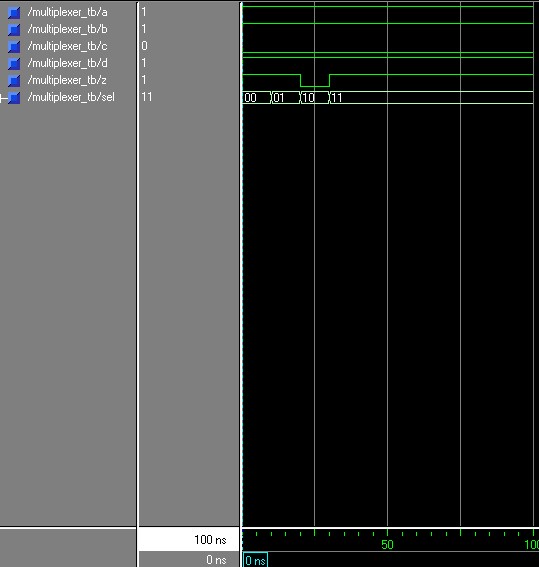
در مقدار دهی اول به تمام پورت ها صفر اختصاص میدهیم تا از حالت unknown خارج شوند .

پس از 10 ثانیه به سیگنال کلاک 1 میدهیم (به ترتیب در بازهای مشخص به کلاک مقادیر مختلف میدهیم .)

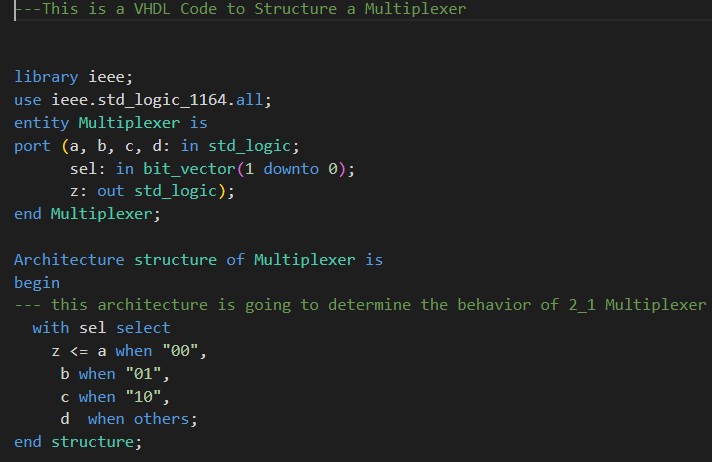
در بازه مختلف کلاک به ریست و D مقادیر مختلف میدهیم .

باید رفتاری مشابه یک فلیپ فلاپ نشان دهد ،

سیگنال ها بصورت زیر خواهد بود :



در مالتی پلکسر 2 به 1 باید با تغییر خطوط سلکتور ، مقادیر خط انتخاب شده به خروجی مالتی پلکسر انتقال بیابد .



در معماری ساختاری مالتی پلکسر ،

پورت ها a, b, c, d از نوع std logic ورودی

پورت Z std\_logic خروجی .

و پورت سلکتور sel آرایه 2 عنصری .

از ساختار with when این پیاده سازی را انجام میدهیم .

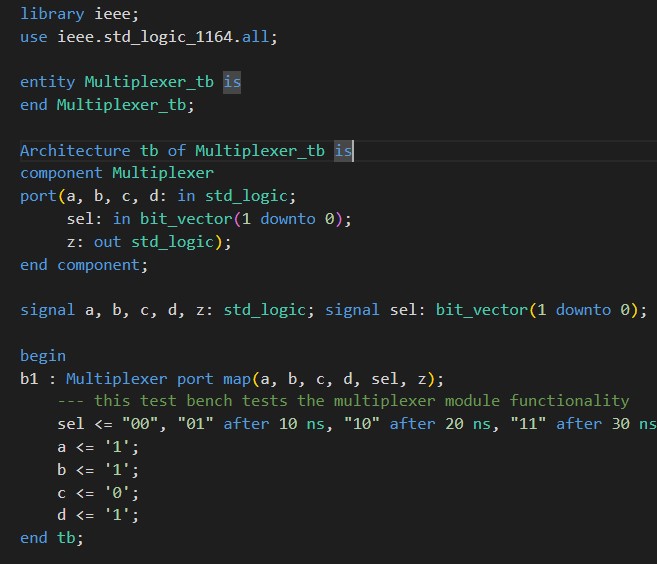
چهار خط انتخاب خواهیم داشت :

* در صورتی سلکتور برابر 00 شد محتوای متغیر a را به z انتقال میدهد
* در صورتی که سلکتور برابر 01 شد محتوای متغیر b را به z انتقال میدهد
* در صورتی که سلکتور برابر 10 شد محتوای متغیر c را به z انتقال میدهد
* در صورتی که سلکتور حالت دیگری گرفت ، d را به z انتقال میدهد .

و در اینجا طراحی ساختار پایان میابد .

در پایان test bench مالتی پلکسر را طراحی میکنیم .

با استفاده از Test bench مطمئن میشویم که شبیه سازی مطابق آنچه از یک مالتی پلکسر انتظار داریم رفتار خواهد کرد .



در این تست بنچ بعد از تعریف سیگنال ها و اضافه کردن کامپوننت multiplexer و مشخص کردن پورت مپ ها که سیگنال ها را به پورت های کامپوننت وصل میکند .

به پورت های a, b, c, d مقادیر دلخواه میدهیم .

و به ازای هر یک از حالت های Selector

سیگنال های خروجی را مشاهده میکنیم و باید طبق حالت سلکتور یکی از چهار ورودی a, b, c, d بر روی z قرار بگیرد .

سیگنال ها به صورت زیر خواهد بود .

