داکیومنت پروژه معماری کامپیوتر

ترم 4032

استاد سمیه کاشی

ابوالفضل شهسواری--------------------402521351

امین شیروانی-------------------------402521369

مستند پروژه: طراحی و تست رجیستر چرخشی در VHDL

**1. مقدمه**

این پروژه مربوط به درس معماری کامپیوتر است و هدف آن طراحی و پیاده سازی یک رجیستر چرخشی (Rotate Register) به زبان VHDL و تست آن با استفاده از یک تست بنچ است.این رجیستر میتواند داده را بصورت چرخشی به چپ یا راست جابجا کند.

**2. محیط توسعه و نصب**

برای پروژه باید از نرم افزار Active-HDL و کتابخانه های IEEE.STD\_LOGIC\_1164 VHDL: استفاده کنیم. اما حین نصب این نرم افزار با چالش فان نحوه ی نصب مواجه شدیم. ابتدا هنگام نصب برنامه هرچقدر اسم نرم افزار را سرچ میکردیم اصلا همچین برنامه ای نبود و بجاش Active.html نمایش داده میشد که در نهایت با کمک gpt تونستیم یک ویدئوی مربوطه پیدا کنیم و نصبش کنیم.در ادامه هنگام نصب برای اضافه کردن Path و تنظیم Key-Value مربوط به برنامه Active-HDL مشکل داشتیم که مجددا gpt به کمک ما اومد.

**3. طراحی رجیستر چرخشی**

در ابتدا با دو بخش اصلی ماژول در VHDL آشنا شدیم.

Entity : تعریف ورودی ها و خروجی ها

Architecture : پیاده سازی منطق ماژول

در بخش Entity، سه ورودی (clk, rst, dir) و یک خروجی (data\_out) تعریف شده است:

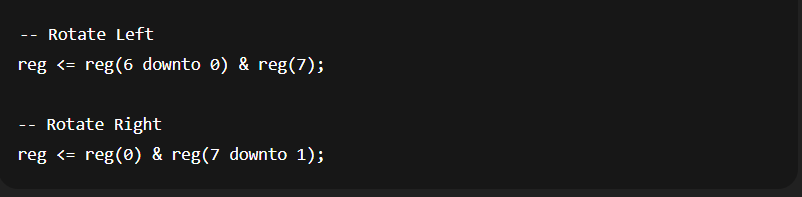
* Clk کلاک
* rstریست
* dir جهت چرخش (0: چپ، 1: راست)
* data\_out: مقدار خروجی رجیستر

در بخش Architecture، از یک سیگنال داخلی reg برای نگهداری مقدار فعلی رجیستر استفاده شد:



در حین تعریف خروجی با مشکل مواجه شدیم. در ابتدا متغیر date\_out را بصورت 8 متغیر STD\_LOGIC جدا تعریف کردیم.این کار باعث شلوغی کد و سختی گسترش آن برای بیت های بیشتر میشد.بجای آن از یک متغیر STD\_LOGIC\_VECTOR برای تعریف خروجی و نوشتن منطق شیفت بر اساس اون استفاده کردیم که خیلی خطوط کمتر و کد امیز تر شد.

در ابتدا منطق شیفت به صورت دستی برای هر بیت نوشته میشد که باعث تکرار و پیچیدگی میشد.که بجای اون از یک نوشتار جمع و جور VHDL برای چرخش استفاده کردیم:



**4. طراحی تست بنچ**

در طراحی تست بنچ ابتدا با استفاده از component ماژول اصلی را وارد تست بنچ کردیم.سپس سیگنال ها را برای اتصال به پورت های ماژول تعریف کردیم و از port map برای اتصال سیگنال های تست بنچ به پورت های ماژول استفاده شد.

در ادامه منطق تست بنپ تلاش کردیم که تمام ورودی ها یعنی clk,rst,dir را در یک حلقه مقدار دهی کنیم ولی کد پیچیده و غیرقابل فهم شد که برای راه حل از دو پروسس جدا استفاده کردیم:

**پروسس اول:** تولید کلاک

**پروسس دوم:** اعمال محرک ها (rst,dir)

در نهایت برای ران گرفتن برنامه هم مجددا سرچ کردیم و متوجه شدیم که باید دو فایل rotate\_register و test bench کامپایل کنیم.سپس برنامه را در حالت شبیه سازی یا simulation قرار دهیم.سپس سلکت کردن متغیر هایی میخوایم بصورت نمودار نشان دهیم و با کلیک راست به waveform اضافه کنیم و در نهایت برنامه را ران کنیم و سیگنال ها و امواج و تغییر اونها را مشاهده کنیم.

**5. نتیجه گیری**

در این پروژه توانستیم:

یک رجیستر چرخشی ۸ بیتی را در VHDL پیاده‌سازی کنیم.

منطق چرخش چپ و راست را به صورت بهینه و کوتاه بنویسیم.

یک تست‌بنچ طراحی کنیم که عملکرد ماژول را بررسی کند.

چالش‌های نصب ابزار، تعریف خروجی، و سازماندهی کد تست را برطرف کنیم.

\*پایان\*