سوال ۸)



الف) در این برنامه ساعت کاری روز اول ساعت ۸ گرفتم.

برای محاسبه زمان من مدول Counter را طراحی کردم که این مدول پارامتر n را

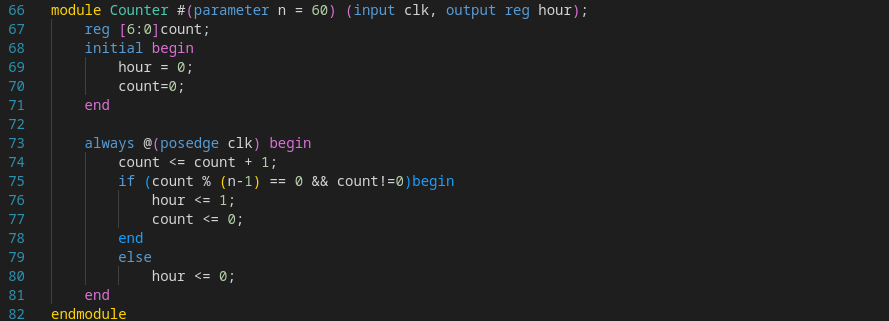
میگیرد ( که مشخص کننده این است که هر چند کلاک را معادل یک ساعت میداند ).

و همچنین سیگنال ورودی clk و سیگنال خروجی hour و متغیر count که در واقع به

نوعی نشان دهنده دقیقه هست برای Counter است.

با متغیر count تعداد کلاک های گذشته را میشماریم و اگر برابر با پارامتر n شد سیگنال

hour را فعال میکنیم و در بقیه حالات hour را با 0 مقداردهی میکنیم.

من هر 60 کلاک را معادل گذشت یک ساعت میدانم.

سپس مدول Parking را طبق خواسته سوال ایجاد کردم همانطوری که

سوال از ما خواسته ورودی ها و خروجی های مدنظر را در پورت های مناسب قرار دادم و

از طرفی برای اعداد ۲۰۰ و ۵۰۰ همان ۹ بیت کافی است. و همینطور ورودی کلاک را نیز

برای مدول Parking در نظر گرفتم و از طرفی ورودی و خروجی در صورت سوال را نیز

در پورت های مناسب قرار دادم. موضوع دیگر در مورد مقدار ساعت و دقیقه است که

برای نمایش ساعت سیگنال timer\_hour را تعریف کردم.

برای نگه داری حداکثر ظرفیت آزاد و دانشگاه دو متغیر university\_capacity و

public\_capacity را تعریف کردیم و برای مقداردهی اولیه دلخواه به آنها از تعریف پارامتر

استفاده کردیم.

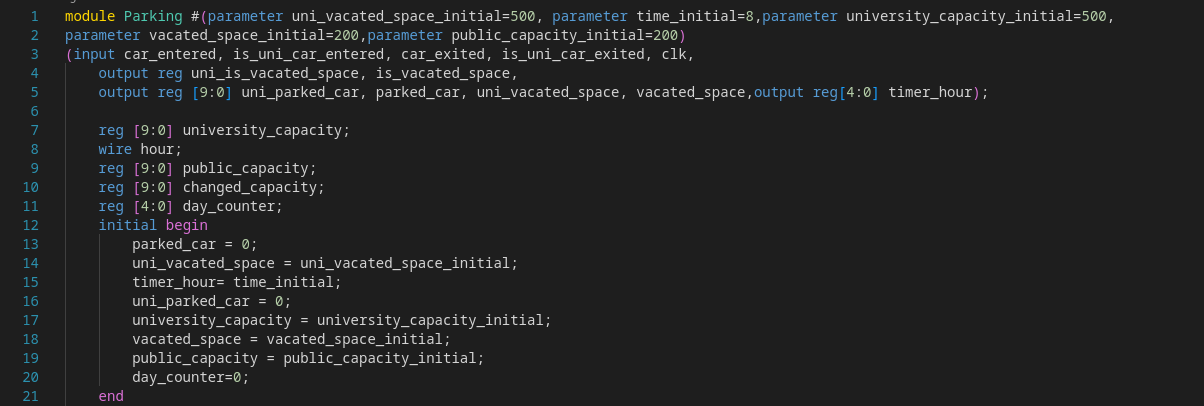
و همچنین برای اینکه بفهمیم ساعت بعدی فرا رسیده ازسیگنال hour استفاده میکنیم

( که به مدول Counter وصل است ).

مقداردهی های اولیه را مطابق خواسته سوال در یک بلاک initial انجام میدهیم.

قسمت پورت و مقدار دهی اولیه سیگنال های خروجی و سیگنال های داخلی به شرح زیر

است:



خب حالا به سراغ توضیح مدول parking و نحوه عملکرد آن میرویم:

در این قسمت یک نکته کلیدی وجود دارد و آن هم این است که همواره اولویت با پرسنل

دانشگاهی است. به این معنا که اگر قرار بر اضافه کردن ظرفیت به قسمت آزاد باشد اگر

به اندازه سقف تعیین شده جای خالی باشد همان مقدار و گرنه به اندازه ظرفیت خالی

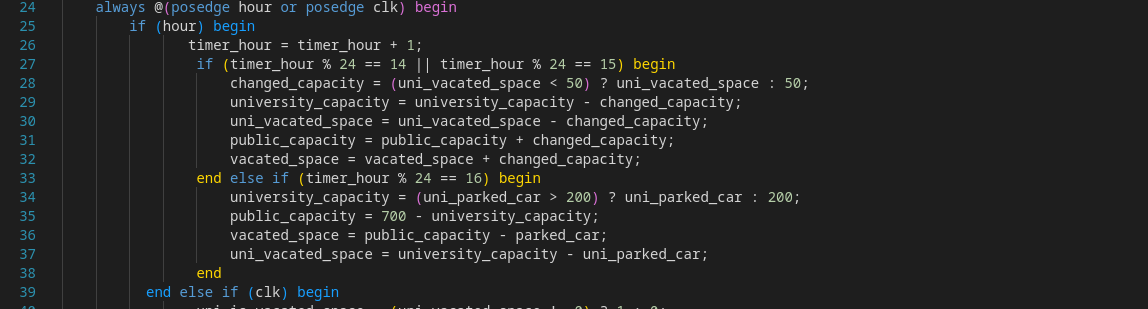
اضافه میکنیم.

نکته دیگر در مورد اضافه کردن ظرفیت است که چون از ساعت ۱۳ به بعد به ازای

گذشت هر یک ساعت ما ظرفیت های آزاد را زیاد میکنیم پس فقط به ازای ساعات ۱۴ و

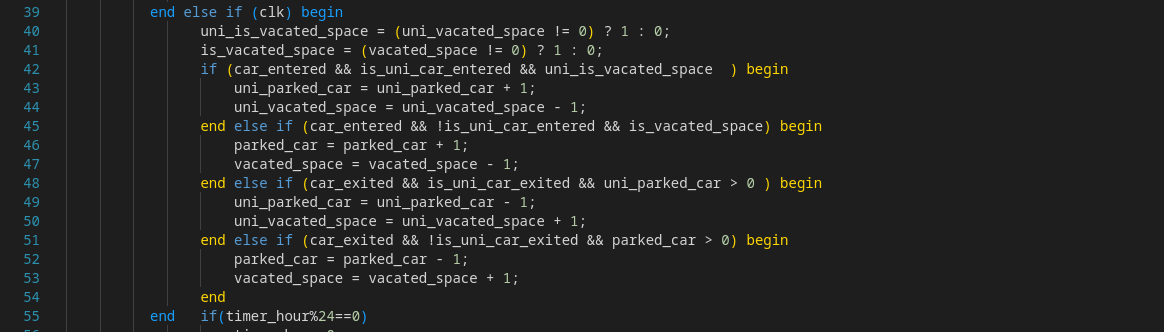
۱۵ ظرفیت تا حداکثر ۵۰ تا زیاد میشود و همچنین در ساعت ۱۶ نیز باید ظرفیت را با

توجه به اینکه دانشگاه اولویت دارد زیاد کنیم. مثلا ببینیم آیا فضای خالی که از دانشگاه

باقی مانده به اندازه ای است که بتوان ظرفیت آزاد را تا 500 تا افزایش داد.

بخش مربوط به ورود ماشین و تغییر ظرفیت آزاد و دانشگاهی نیز که با سیگنال clk کار

میکند به شکل زیر میباشد:



در این قسمت نیز مجددا بحث اولویت ها و اینکه من باید ببینم اصلا ظرفیت دارم که

ماشین اضافه کنم را مشاهده میکنیم و یه موضوع دیگر این که برای من در موقع تست

گرفتن پیش آمد این بود که ممکن است سیگنال خروج ماشین زیاد فعال بماند و تعداد

ماشین های باقی مانده به حدی کم شود که عدد حاصل منفی شود و من از این رو، شرط

مثبت بودن تعداد ماشین های پارک شده را چک کردم. نکته دیگر چون من میخوام ساعت

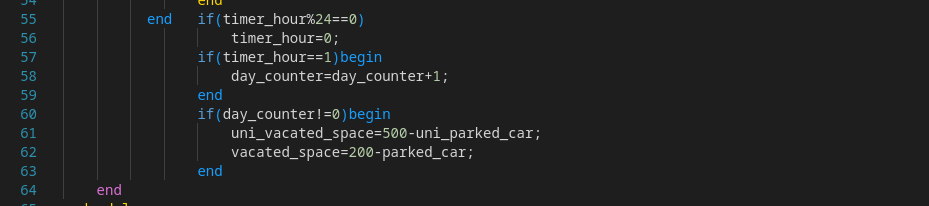
را هم نمایش دهم چک میکنم که اگر ساعت ۲۴ شد، به صفر تغییر کند.

حرکت دیگه که من کردم در مورد این بود که سیستم برای روز دوم هم به درستی کار کند.

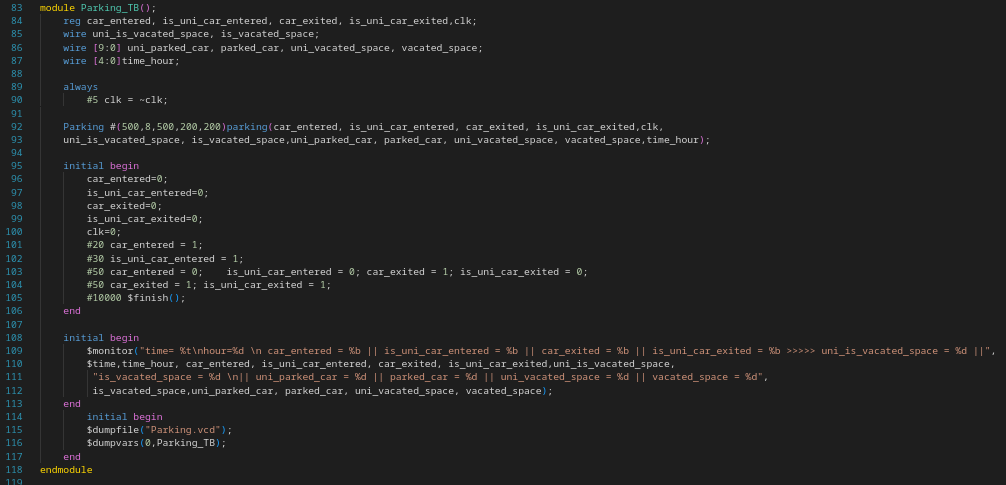
از این رو متغیر day\_counter را تعریف کردم که با یک شدن متغیر hour\_counter

کار میکند و هر موقع که این اتفاق بیفتد ظرفیت ها بر اساس تعداد ماشین های درون

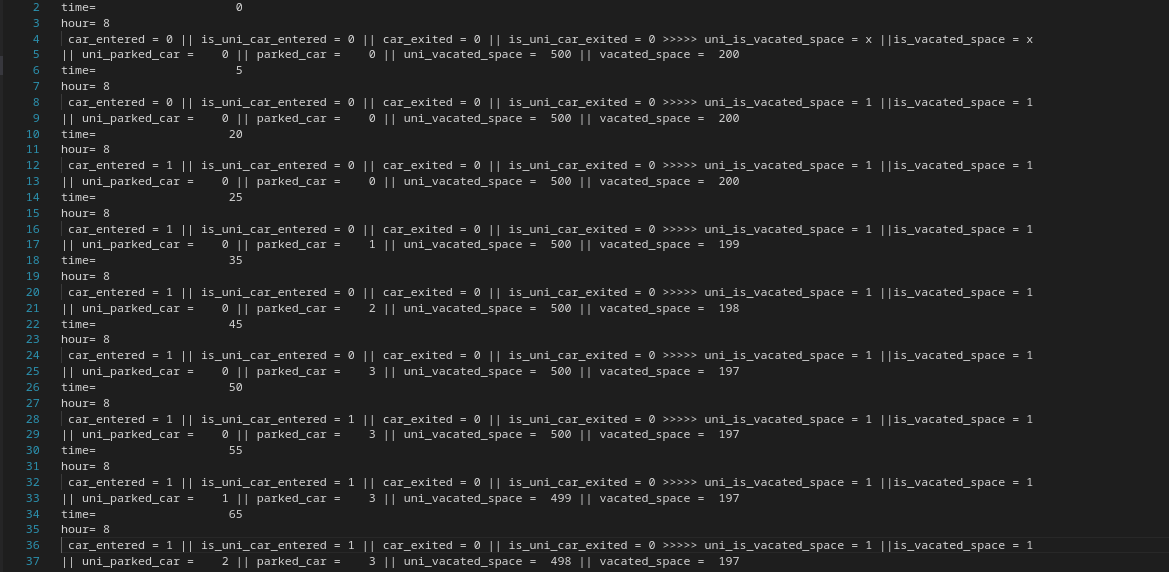
پارکینگ به حالت اولیه برمیگرداند.



حال به سراغ testBench نوشتن میرویم. تست بنچ من به شکل زیر است و خروجی های

آن نیز در فایل output.txt آورده شده است:

بخشی از خروجی این testBench به شکل زیر است:



اگر بخواهیم کمی خروجی را تحلیل کنیم تا از عملکرد درست آن مطمئن شویم:

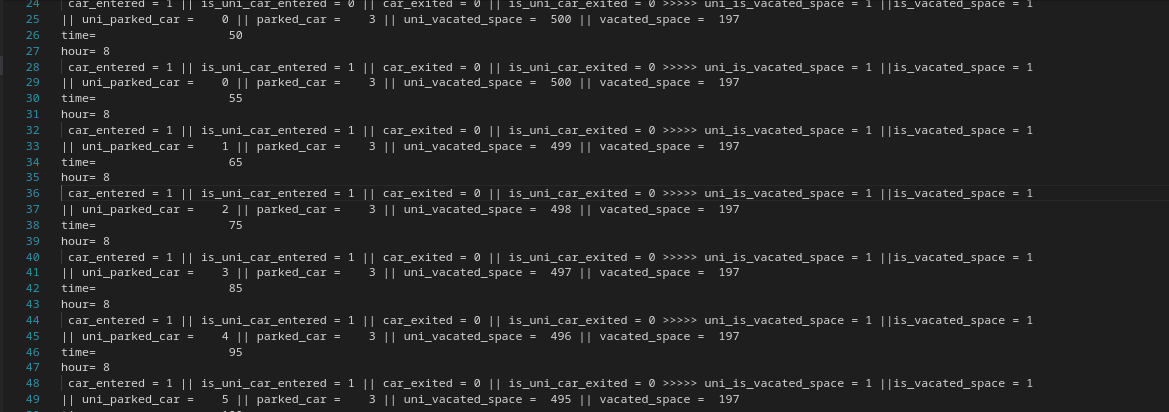
در ساعت ۸ که سیستم شروع به کار میکند باید ظرفیت های دانشگاه و آزاد به ترتیب 500 و 200 باشد که

همانطور که میبینیم اینگونه است.

در زمان 20 یک ماشین آزاد وارد میشود که نتجیتا در کلاک بعدی باید ظرفیت آزاد یکی کم شده باشد، که همینطور نیز است.

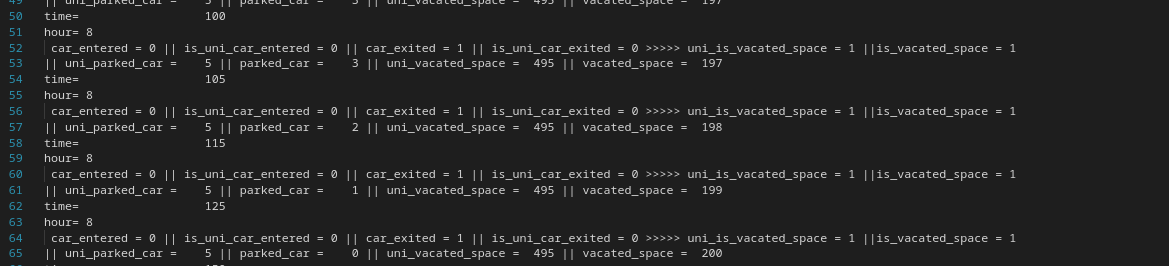
این روند ادامه دارد و مدام بر تعداد ماشین های آزاد افزوده میشود تا اینکه در زمان 50، نوع این

ماشین عوض میشود و ماشین پرسنلی وارد میشود از این زمان تا زمان 100، مدام ماشین

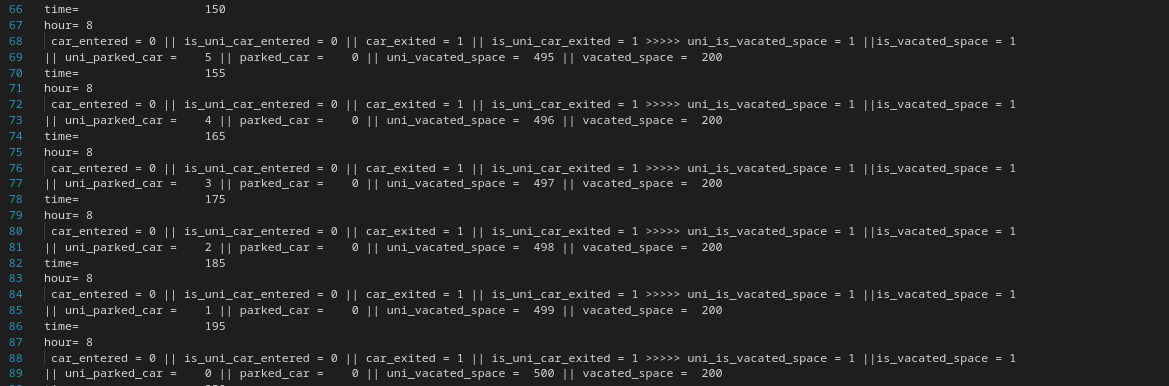
پرسنلی وارد میشود 

و همانطور که میبینیم از اینجا به بعد دیگر ماشینی وارد نمیشود و تا زمان 125 ما با خروج ماشین

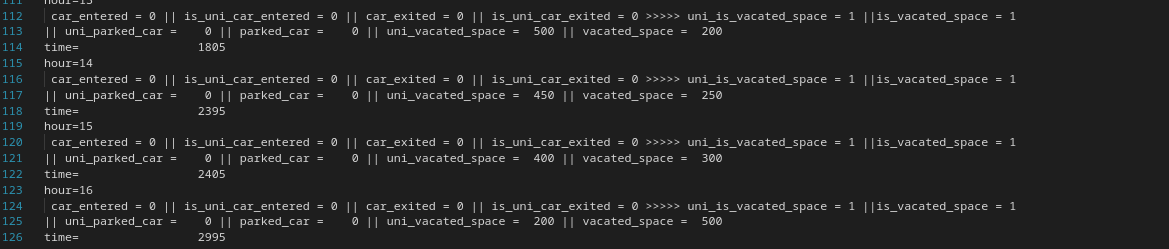
های آزاد روبرو هستیم.



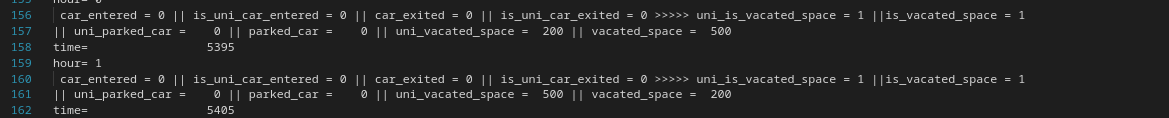
در زمان 150 ماشین های دانشگاهی شروع به خروج میکنند و این روند تا زمان 250 ادامه

خواهد داشت:

از نکات دیگر طراحی میتوان به اینکه در ساعت 14 و 15 و 16 باید ظرفیت آزاد زیاد شود اشاره کرد که در ادامه میبینیم:

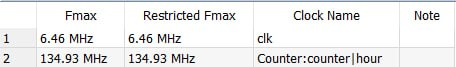


حالا در آخر میتونیم اون قضیه ست شدن ظرفیت ها را برای روزهای بعدی چک کنیم:



ب) فایل مربوط به Quartus این پروژه در فولدر پروژه آمده است.

همانطور که در تصویر زیر میبینیم بیشینه فرکانس برابر با 6.46MHz است.



سیگنال clk همان سیگنال کلاک مدول Parking است. ما میتوانیم درستی این فرکانس را با مراجعه

کردن به گزارش تاخیر مسیر داده ها چک کنیم:



به طور کلی تاخیر مسیر داده ها برابر با 154ns است و همانطور که میبینیم فرکانس بیشینه تقریبا با

معکوس آن برابر است:

1/154ns~6.49MHz