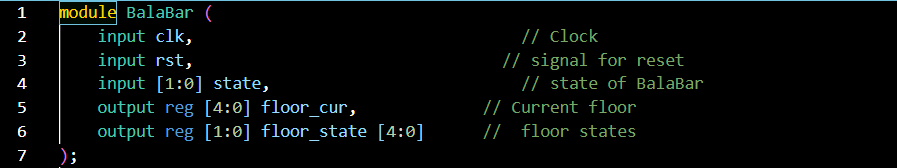
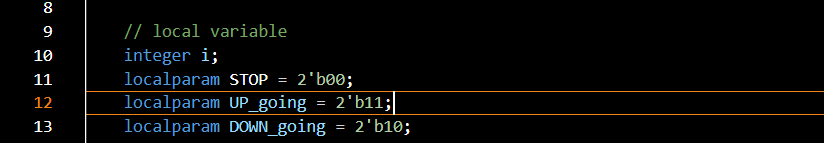
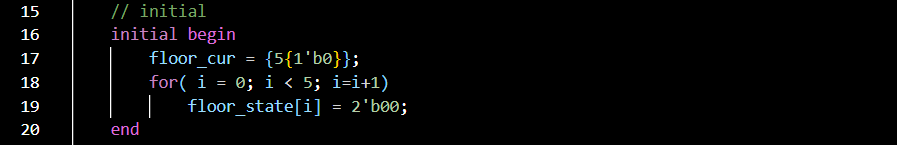
در این سوال ما باید عملکرد آسانسوری را شبیه سازی کنیم.که شامل طبقه همکف و 4 طبقه دیگر است.در نظر داریم که هر طبقه یک کلید در خارج طبقه و یک کلید در داخل است.دو فایل ساخته که یکی همان بالابر و دیگری میشود کترلر آن.  
  
ابندا ماژول بالابر را پیاده سازی میکنیم.

# ماژول بالابر:



سه سیگنال ورودی دارد که یکی کلاک است و دیگری سیگنال ریست و دیگری سیگنال وضعیت که در هر لحظه وضعیت بالابر را نشان میدهد. همچنین در خروجی نیز یک سیگنال که طبقه فعلی آسانسور و همچنین یک آرایه برای تعیین وضعیت طبقات است. مثلا 3 یعنی توقف آسانسور در آن طبقه و یا 2 یعنی خانه فعلی آسانسور اما متحرک و 1 یعنی خانه قبلی آسانسور اما متحرک.  
  


متغیر های داخلی را داریم که شامل i برای حلقه ها و همچنین سه متغیر تعریف شده برای وضعیت آسانسور که 00 به معنی توقف و 10 به معنی حرکت به سمت پایین و 11 به معنی حرکت به سمت بالاست.  
  


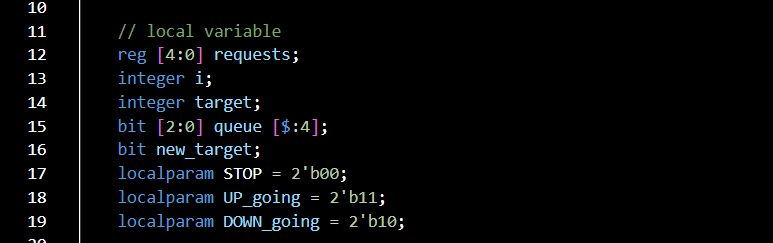
در بخش اولیه ابتدا وضعیت تمامی طبقات برابر 00 میگذاریم که یعنی عدم حضور آسانسور.همچنین طبقه فعلی را هیچکدام نمیگذارد.



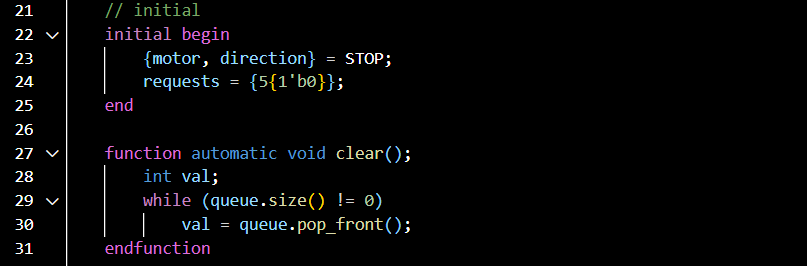
بخش اصلی این ماژول که با 1 شدن کلاک و یا 1 شدن ریست فعال میشد.اگر ریست فعال بود که طبقه فعلی را همان همکف میگذارد و همچنین همیشه ابتدا وضعیت تمامی طبقات را برابر 00 میکند.  
سپس با توجه به نحوه حرکت وضعیت طبقات را بروزرسانی میکند.برای مثال اگر در وضعیت ایستادن بود طبقه فعلی را برابر 11 میکند و یا اگر در حرکت بود با توجه به جهت حرکتش طبقه قبلی را 01 و طبقه جدید را 10 میکند.  
  
این ماژول در واقع حرکت بالابر را شبیه سازی میکند و برای نمایش خروجی به کار می رود.

# ماژول کنترلر :

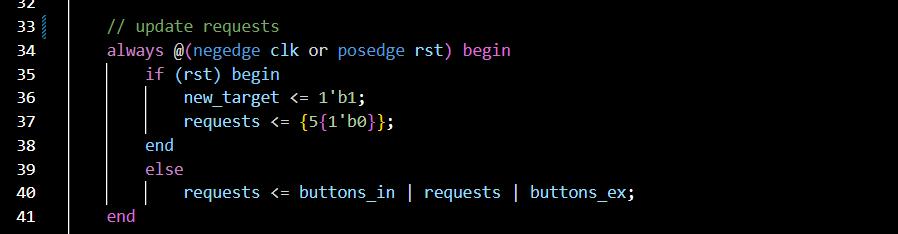
پورت های ورودی شامل کلاک و ریست و همچنین طبقه فعلی آسانسور و همچنین کلید های داخلی و خارجی آسانسور میشود.که این ها هر بیتشان که 1 شد یعنی برای آن طبقه درخواست وجود دارد. همچنین دو تا بیت خروجی که یکی نشان دهنده آن که آیا اصلا حرکت میکند یا ساکن است که این همان motor است و دیگری direction میباشد که اگر 1 بود یعنی حرکت به بالا و 0 یعنی حرکت به پایین.

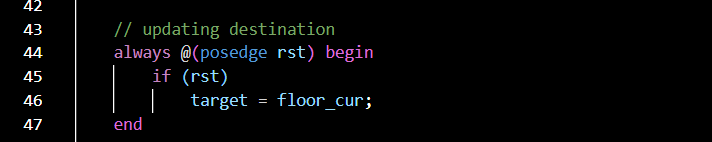


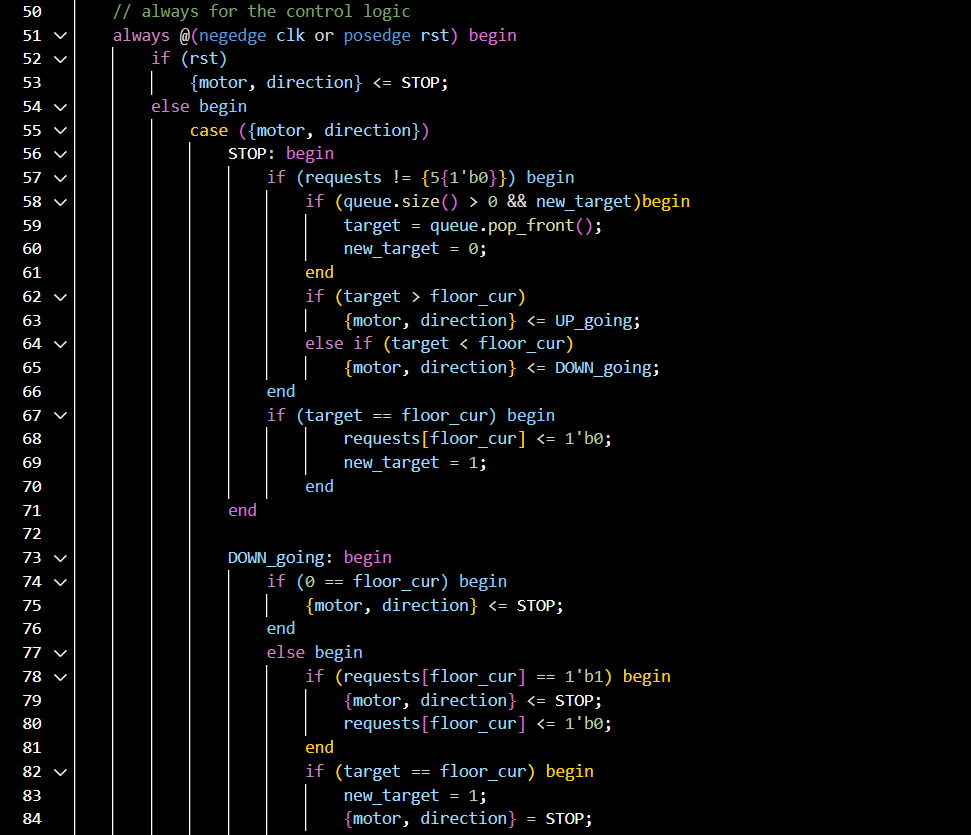
متغیر های داخلی که مانند ماژول قبلی شامل i و متغیر های داخلی STOP و ... میباشد. همچنین شامل صف برای رعایت ترتیب و متغیری برای درخواست مقصد جدید و یک آرایه که بیت مد نظر آن اگر 1 بود یعنی برای آن طبفه در خواستی وجود دارد.

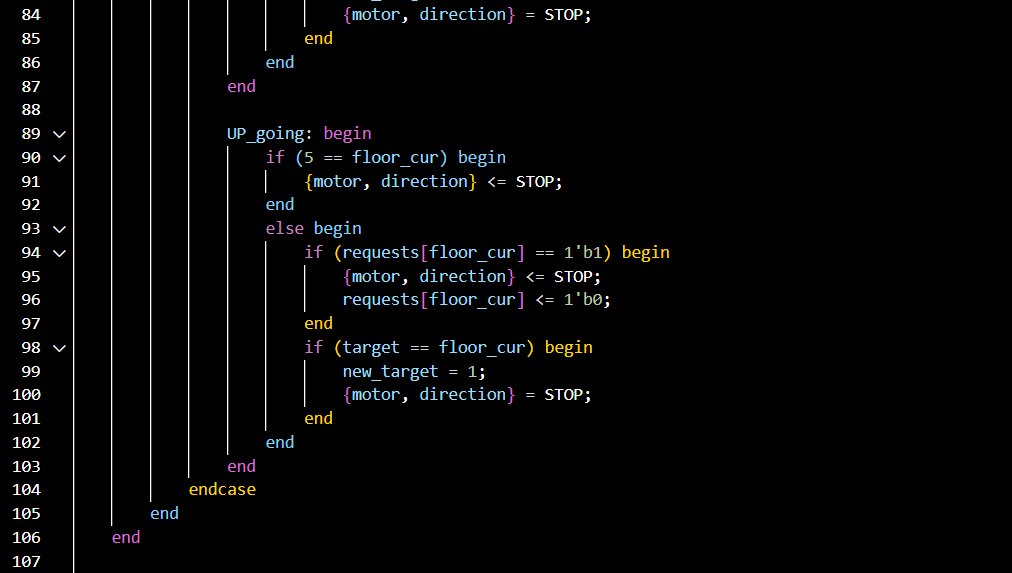


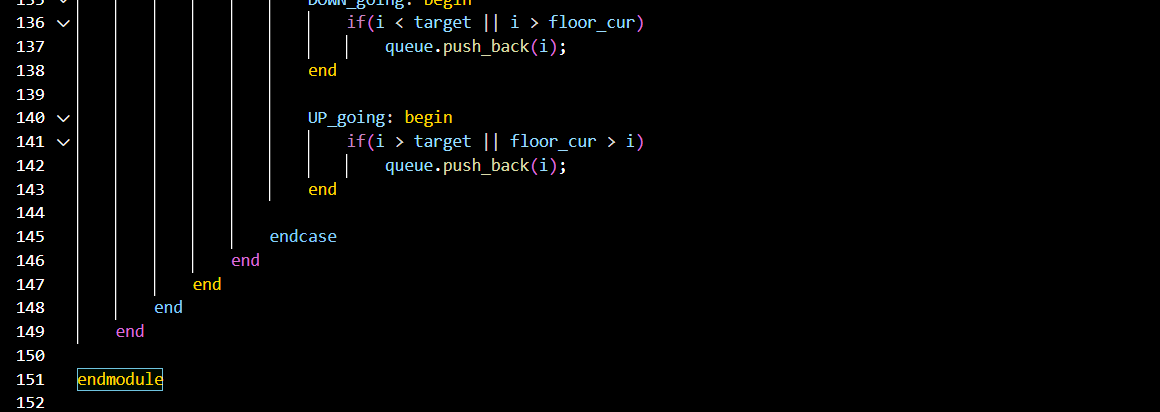
تعریف اولیه که آمده و تمامی درخواست هارا برابر 0 گذاشته و وضعیت توقف را برای بالابر در نظر دارد.  
همچنین در بخش پایین شاهد یک تابع هستیم که مسئول خالی کردن صف است.



در این بخش اگر ریست فعال بوده ریست میکنیم و تمامی درخواست ها برابر 0 میشوند و همچنین مفدار درخواست هارا با توجه به وضعیت کلید های درونی وبیرونی معلوم میکند.

در این بخش نیز در صورت ریست کردن هدف را برابر طبقه فعلی میگذارد.



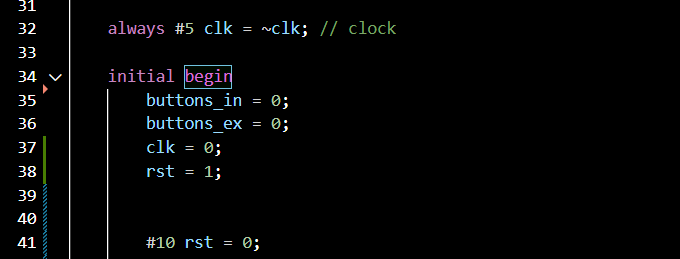
در این بخش به درخواست ها رسیدگی میشود با توجه به وضعیت موتور و وضع درخواست ها.اگر نیاز است که به صف و و الا در مسیر به آن طبقه میپردازد.

در بخش پایانی نیز با توجه به حرکت موتور و طبقه آن دستور میدهد که بالابر چگونه حرکت کند.

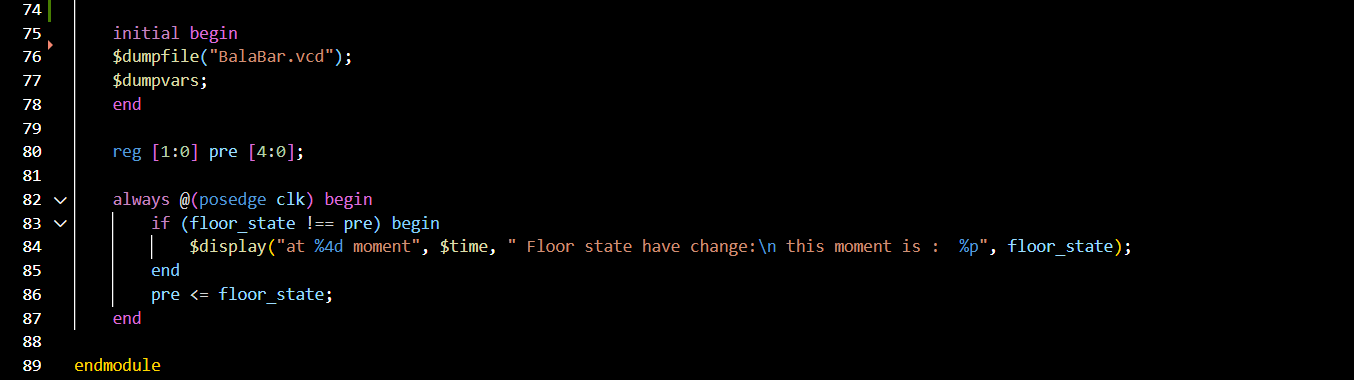
# تست



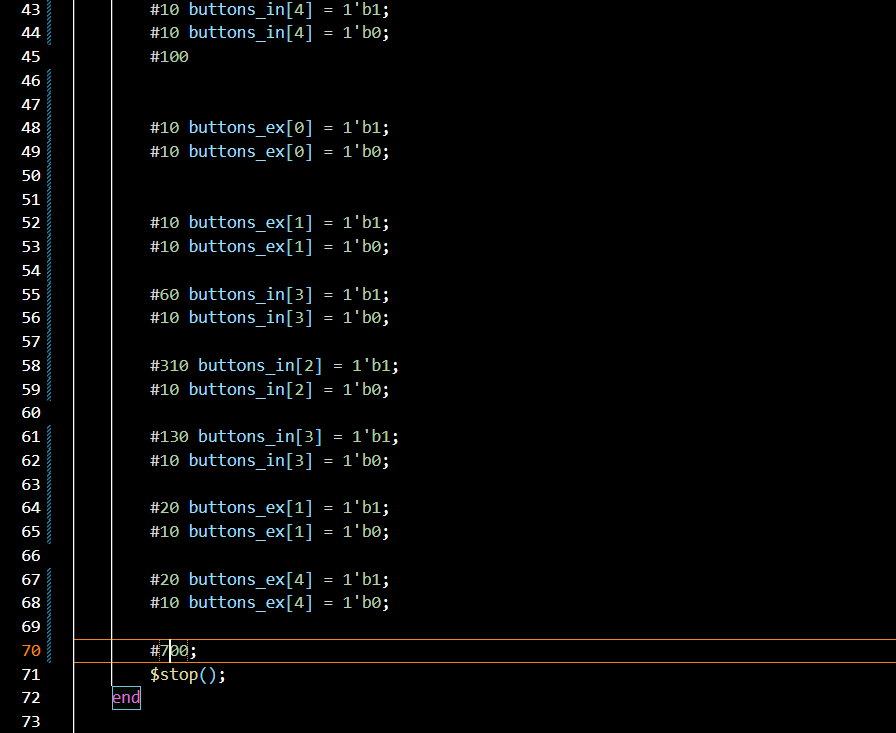
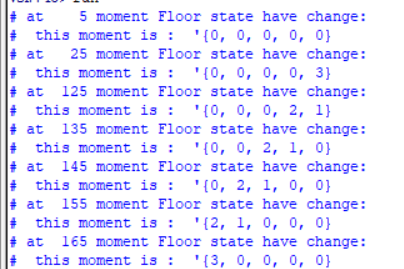
در این بخش ما چند متغیر داخلی که همان ورودی های دو ما/زول قبلی اند را ساخته و یک instance از هر کدام آن ها میگیریم.



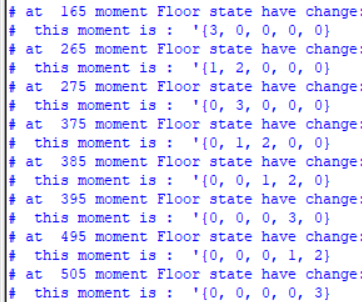
در اینجا نیز ابتدا کلاک را هر 5 ثانیه یکبار تاگل میکنیم و در 10 ثانیه اول ریست میکنیم.



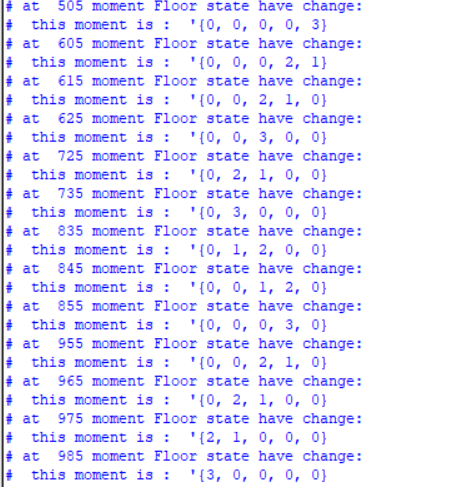
این بخش هم ساز و کاری است برای نمایش خروجی و به محض تغییر وضعیت طبقات وضعیت هر طبقه را پرینت میکند.

حال به سراغ سناریوی جالبمان میرویم.  
در خط 43 کلید طبقه 4 میخورد.  
مشاهده میشود که بالابر پس ا زتوقف در طبقه 1 (به دلیل ریست) مستقیم به طبقه 4 میرود و در آنجا توقف میکند:  


سپس در حین همین حرکت در خطوط 48 الی 56 مشاهده میشود که کلبد طبقات 3 و 1 و همکف میخورد. با اینکه اول کلید همکف میخورد اما چون طبقات دیگر در مسیر هستند پس در آن جا نیز توقف میکند.



در خط 58 درخواست طبقه 2 می آید که در حینش در خواست طبقه 3.  
بهد از آن هم درخواست طبقه همکف و بعد از آن درخواست طبقه 4. اما مشاهده میشود که بالابر بعد از رفن به طبقه دو و بعد ار آن به سه، مستقیم سراغ نمیرود و اولویت دارد که کدام زود تر زده و به یک میرود و سپس بعد از آن به طبقه 4 میرود.



مشاهده شد که در تست تمامی حالات اعم از وسط راه وایسادن، اولویت در نظر داشتن، منطق درست، عدم تخطی از 5 طبقه مجاز و... به درستی عمل میکردن.