به نام خدا

امیر علی واشقانی فراهانی شماره دانشجویی:99102489

در این پروژه، یک مدار کنترل آسانسور برای یک ساختمان چهارطبقه (همکف + 4 طبقه) طراحی و پیاده‌سازی شد. این مدار قادر است به درخواست‌های کاربران در هر طبقه و داخل آسانسور پاسخ دهد و به‌صورت بهینه حرکت آسانسور را کنترل کند.

2. اهداف

طراحی و پیاده‌سازی یک مدار منطقی برای کنترل حرکت و توقف آسانسور.

مدیریت درخواست‌های کاربران به‌صورت اولویت‌دار و بر اساس زمان فشرده شدن دکمه‌ها.

شبیه‌سازی و تست عملکرد مدار در سناریوهای مختلف.

3. تعریف ورودی‌ها و خروجی‌ها

ورودی‌ها:

حسگر هر طبقه (5 حسگر)

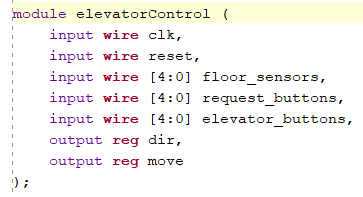
دکمه درخواست طبقات (5 دکمه)

دکمه‌های داخل آسانسور برای تعیین مقصد (5 دکمه)

خروجی‌ها:

جهت حرکت (بالا/پایین)

وضعیت حرکت (توقف/حرکت)

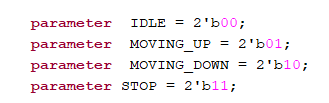


4. تعریف حالت‌ها

حالت توقف در هر طبقه

حرکت به سمت بالا

حرکت به سمت پایین



قسمت IDLE برای تصمیم گیری جهت و حرکت کردن و ایستادن اسانسور میباشد که هنگام شروع به کار اسانسور وارد ان مرحله میشویم بقیه مراحل نیز با توجه به اسم ان ها مشخص میباشد .

5. مدیریت درخواست‌ها

برای مدیریت درخواست‌ها از یک صف اولویت (FIFO) استفاده شد. دلیل انتخاب این روش، ساده و مؤثر بودن آن برای نگهداری درخواست‌ها به ترتیب زمان فشرده شدن دکمه‌ها بود. صف اولویت به‌صورت حلقوی (circular queue) پیاده‌سازی شد تا از بروز خطاهای پر شدن صف جلوگیری شود

tail <= (tail + 1) % 10;.

6. پیاده‌سازی

پیاده‌سازی در زبان Verilog انجام شد و شامل دو بخش اصلی بود:

مدار کنترل آسانسور:

مدیریت حالت‌های آسانسور با استفاده از یک ماشین حالت متناهی (FSM).

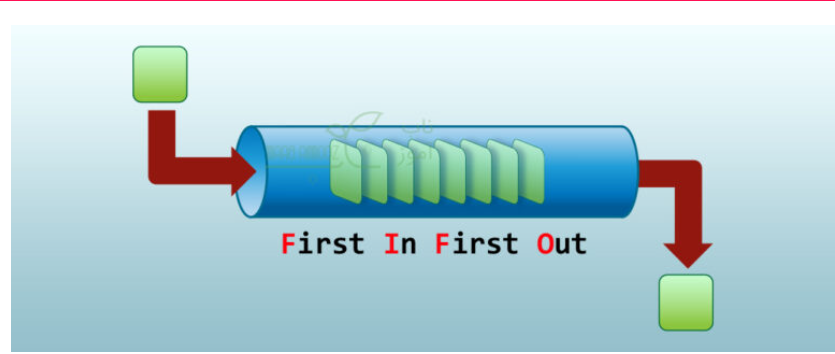
مدیریت صف اولویت برای نگهداری درخواست‌ها.

کنترل حرکت و توقف آسانسور بر اساس درخواست‌ها.

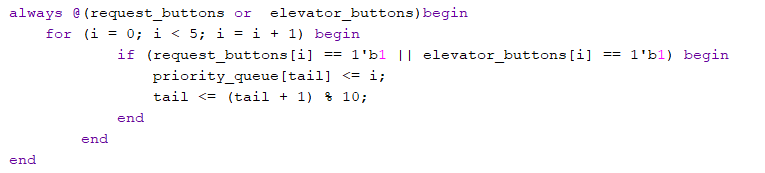
**حافظه های FIFO**

نوع دیگری از ساختارهای حافظه که نقطه مقابل حافظه های LIFO است، حافظه FIFO می باشد. برای تجسم عملکرد آن می توانید یک صف از افراد را در نظر بگیرید. افراد به ترتیب وارد صف می شوند. به مرور طول صف افزایش می‌یابد. سپس افراد به ترتیبی که وارد صف شده اند، از صف خارج می‌شوند. در این صورت اولین نفری که وارد صف شده، اولین نفری است که از صف خارج خواهد شد. به همین دلیل به این نوع حافظه First In First Out و یا به اختصار FIFO می‌گویند. تصویر زیر نمایی از ساختار حافظه FIFO را نمایش می‌دهد.

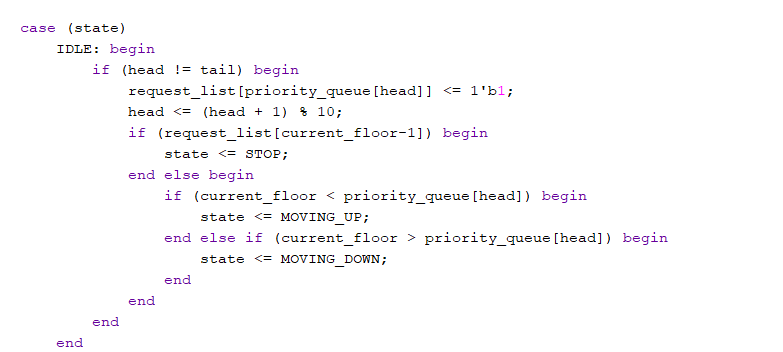
[برنامه نویسان](https://fa.wikipedia.org/w/index.php?title=%D8%A8%D8%B1%D9%86%D8%A7%D9%85%D9%87_%D9%86%D9%88%DB%8C%D8%B3%D8%A7%D9%86&action=edit&redlink=1) و کاربران صف FIFO بایستی توجه زیادی به کاربرد واژگان head و tail برای اشاره به دو سر صف داشته باشند



**در این پروژه ابتدا ما یه حافظه طراحی کردیم که مانند fifo میباشد از این حافظه برای تعیین اولویت ها استفاده کردیم که اگر دستور حرکت به سمت بالا امد و پس ان دستور حرکت به سمت پایین امد ابتدا به سمت بالا و سپس به سمت پایین حرکت کند**

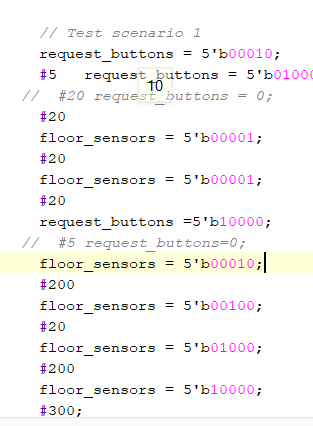
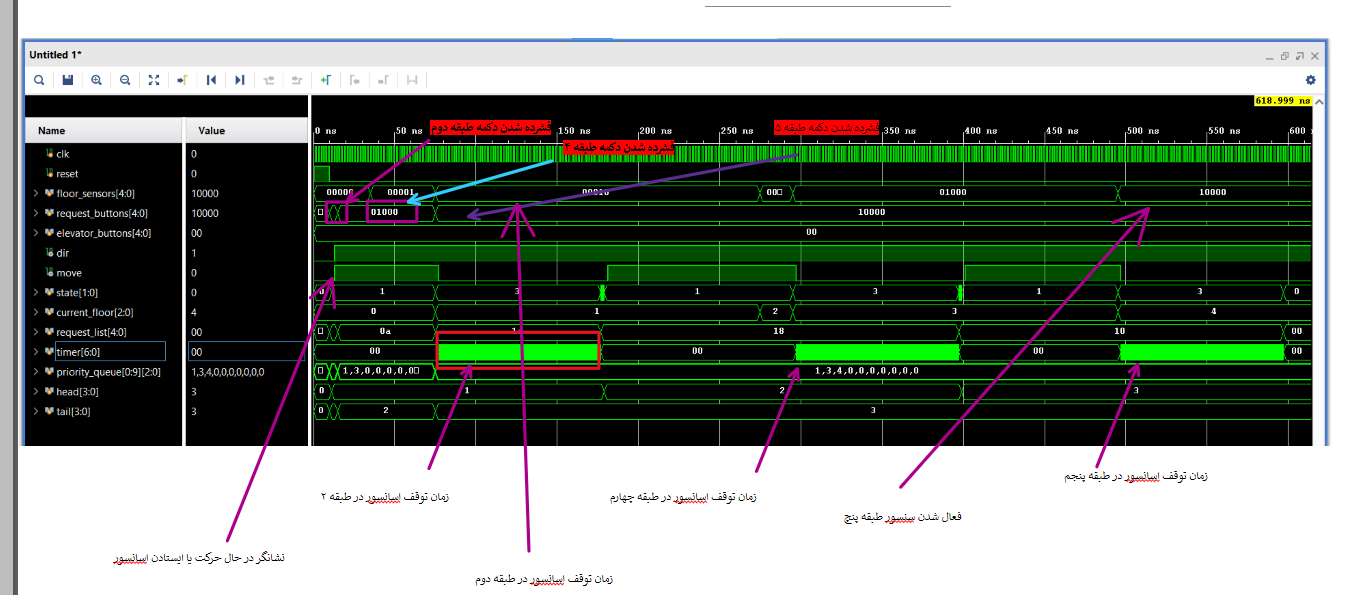
****

**همانطور که مشاهده میشود این بخش کد ما حساس به کلید های داخل و بیرون اسانسور است که با هر بار فشردن انها دستور جدید وارد حافظه میشود**

****

**سپس یک مرحله برای بررسی وضعیت اسانسور مینویسیم که مشخص میکند این اسانسور به سمت بالا برود یا پایین یا متوقف شود**

**همچنین در اینجا هم یک متغیر request\_list نوشتیم که تمام دستور های کلید های اسانسور به ان منتقل میشود برای مثال وقتی فقط کلید طبقه دو فشرده شده باشد این متغیر دارای ارزش 00010 میباشد و وقتی که در حین حرکت و هنوز به طبقه مورد نظر نرسیده باشد و کلید دیگری مثلا کلید طبقه 4 فشرده شود این متغیر request\_list دارای ارزش 10010 میشود که به این معنی میباشد که باید در طبقات دو و چهار متوقف شود**

****

**همچنین در مدار کنترل این اسانسور چون از fifo استفاده شده است اگر دکمه طبقات بالا و پایین با فاصله کوتاهی از هم زده شوند با توجه به ترتیب فشرده چون دکمه ها اسانسور حرکت میکند .**