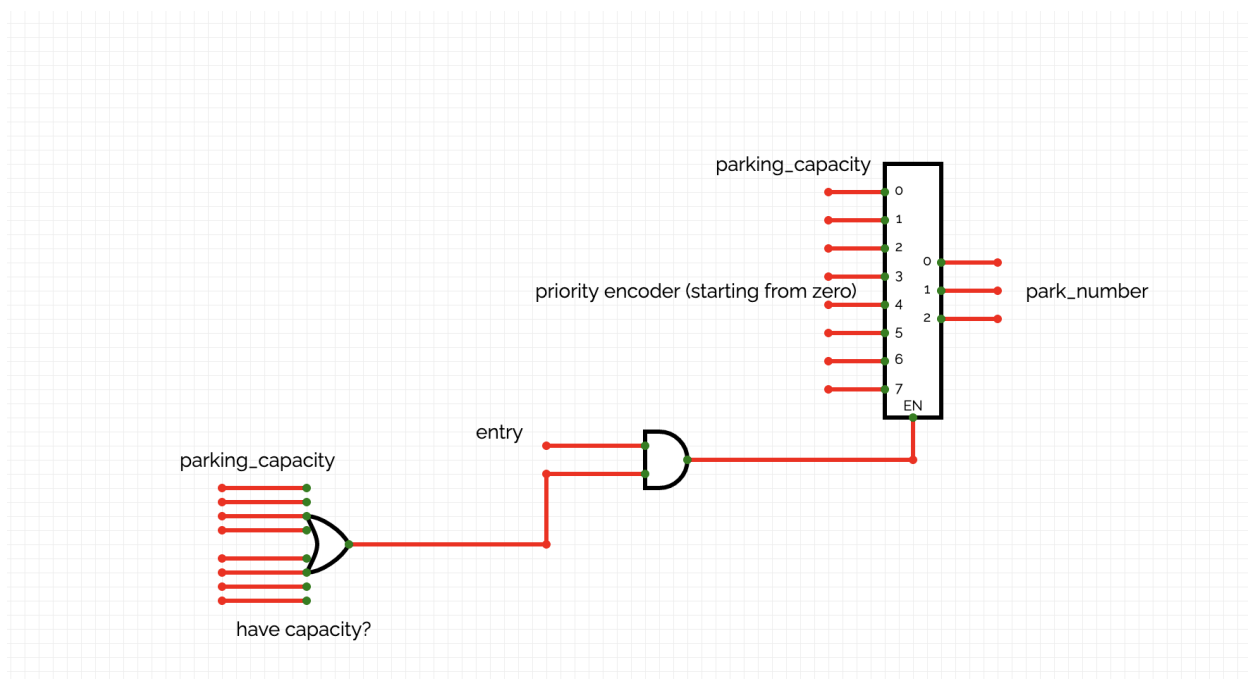


ماژول 1

در این ماژول باید برای خودروی وارد شده یک جای پارک پیدا کنیم. طراحی ساختاری این ماژول به صورت زیر است.

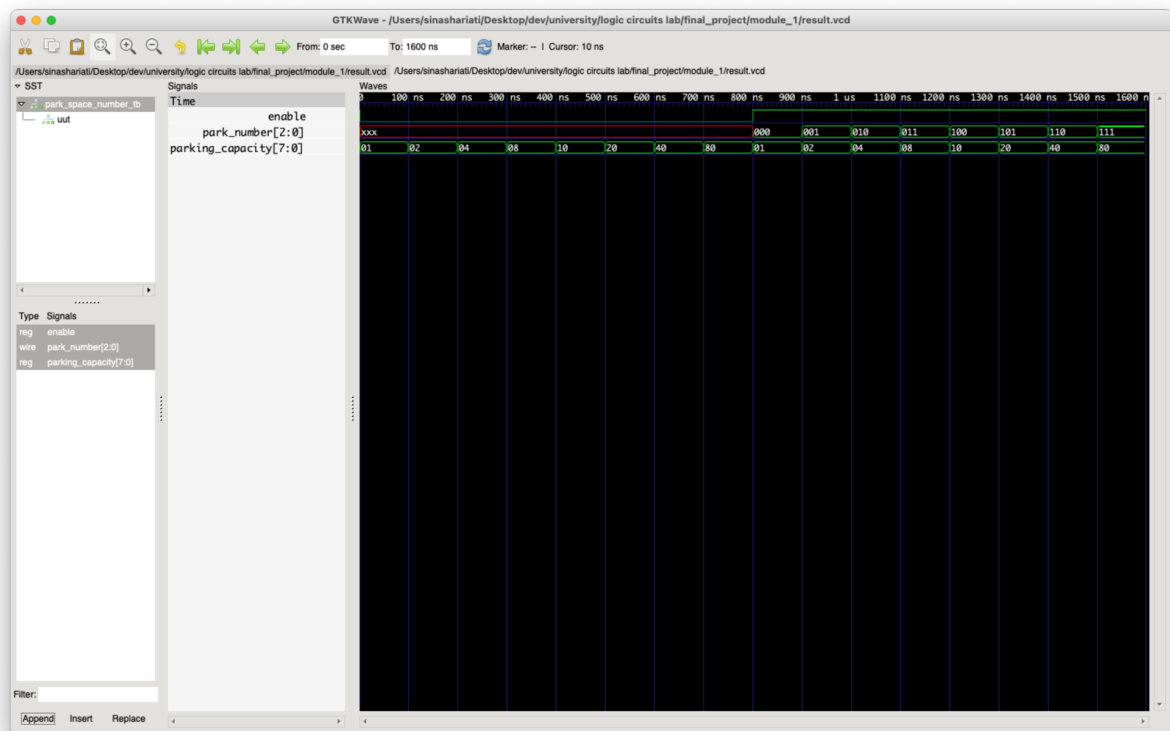
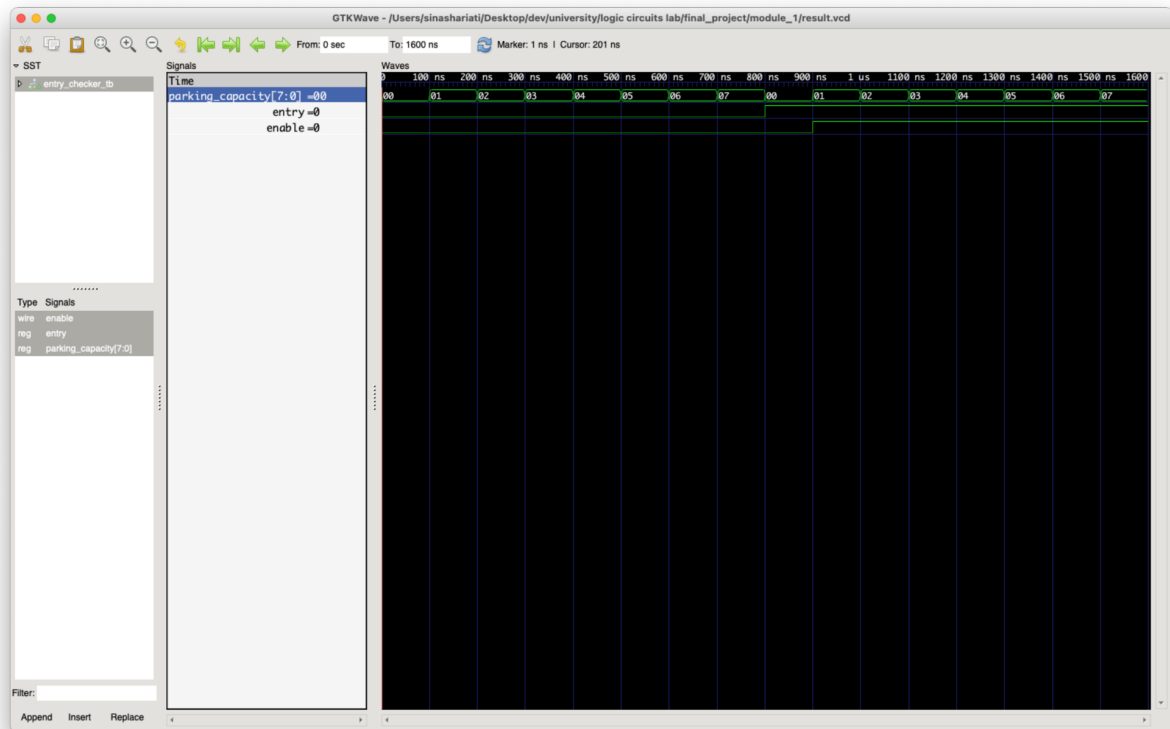


که به این صورت کار میکند که:

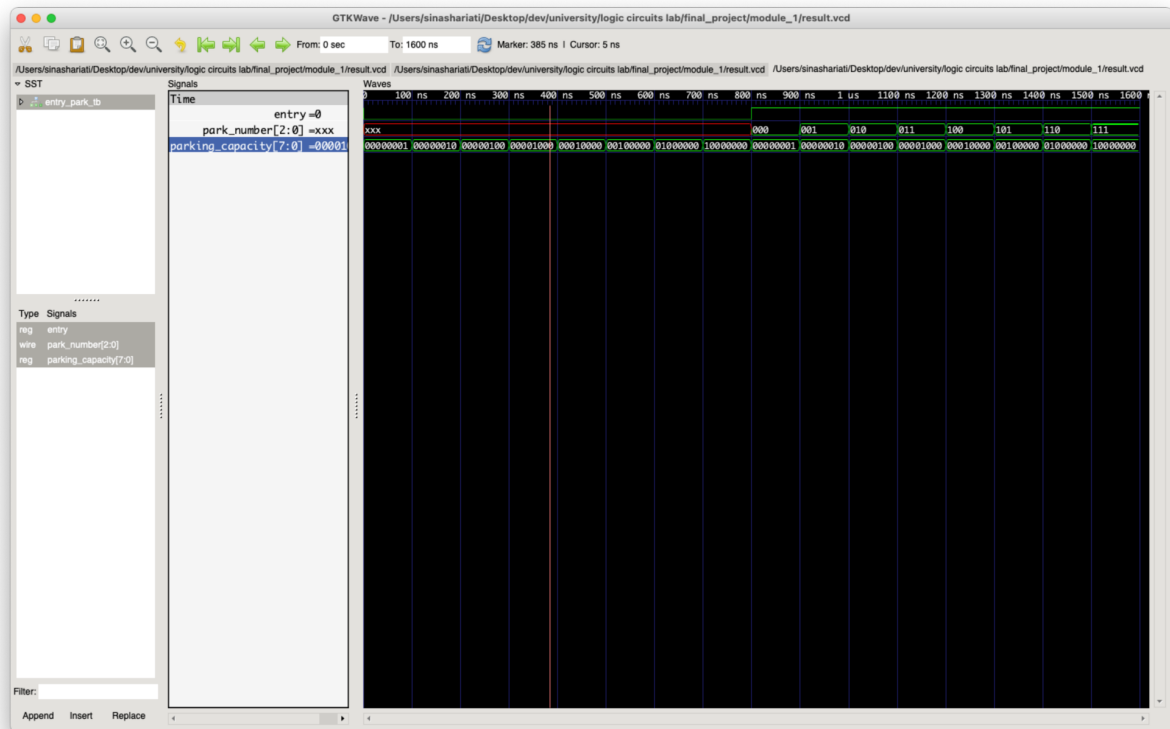
ابتدا منتظر می شود که یک ماشین وارد شود و اگر وارد شود **entry** به صورت **active** در می آید. سپس میدانیم که دستور کار پروژه آمده است که **parking capacity** یک **vector** است که 8 بیت دارد و هر بیت اگر 1 باشد نشان دهنده ی خالی بودن آن جای پارک است. پس اگر یکی از بیت ها 1 باشد یعنی پارکینگ هنوز جا برای پارک ماشین جدید دارد که این را با استفاده از **or** مشخص شده در طراحی می توان متوجه شد. حال که ماشین وارد شده و ما نیز برای آن جا داریم باید آن را به نزدیکترین جای ممکن ببریم که برای این کار از یک **priority encoder** باید استفاده کنیم. حال چون هیچ **testbench** یی به ما داده نشد و نه در دستورکار نه توسط مسئول آزمایشگاه یک حالت خاصی مطرح نشد ما فرض میکنیم که هر چه ارزش بیت کمتر باشد ارزش بیشتری دارد. پس **priority encoder** ما در واقع **low bit priority encoder** است و ماشین ها را از 0 تا 7 به ترتیب پخش می کند.

برای اینکه **entry park** درست کار کند باید به عنوان **top level module** عمل کند برای **entry checker** و **park space number** عمل می کند.

حال اگر **testbench** های مربوط به **entry checker** و **park space number** را به ترتیب **run** کنیم خروجی های زیر را خواهیم داشت :



توجه داشته باشید که در مازول park space number زمانی که enable یک نشده باشد نباید هیچ خروجی مشخصی بدهد پس خروجی آن به صورت xxx در نظر گرفته شده است.
حال برای entry park داریم که:



و دلیل وجود xxx در ان به این دلیل این است که اصلا ورودی ماشین نداشته ایم پس نباید هیچ park number یی را تولید کنیم.