

بسمه تعالی



پیش گزارش آزمایش سوم آزمایشگاه مدارهای منطقی

تابستان ۱۴۰۳

## شمارنده ها

استاد

شاهین حسابی

اعضای گروه:

محمد مهدی عابدینی ۴۰۲۱۰۶۱۹۱

امیر مهدی وزیری ۴۰۲۱۰۶۷۵۶

سید محمد رضا جوادی ۴۰۲۱۰۵۸۶۸

فهرست عناوین

۲	.....مقدمه
۳	.....آزمایش اول
۳	.....بخش اول :
۶	.....بخش دوم
۶	.....آزمایش دوم

## مقدمه

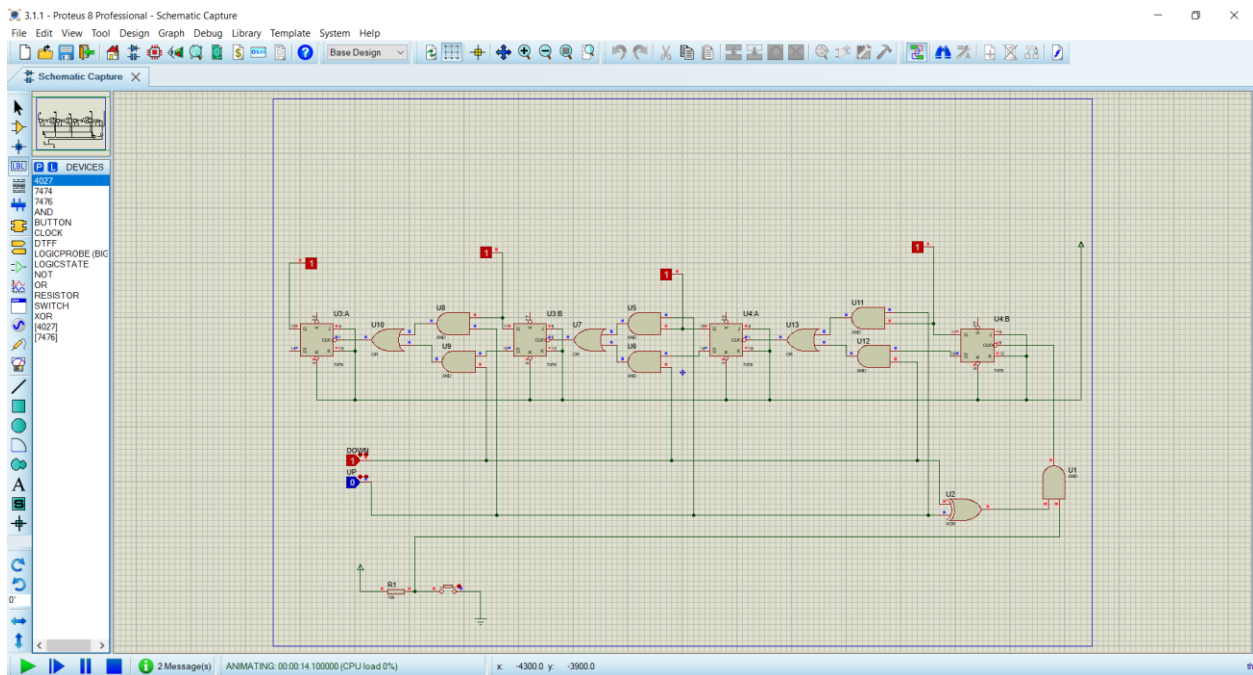
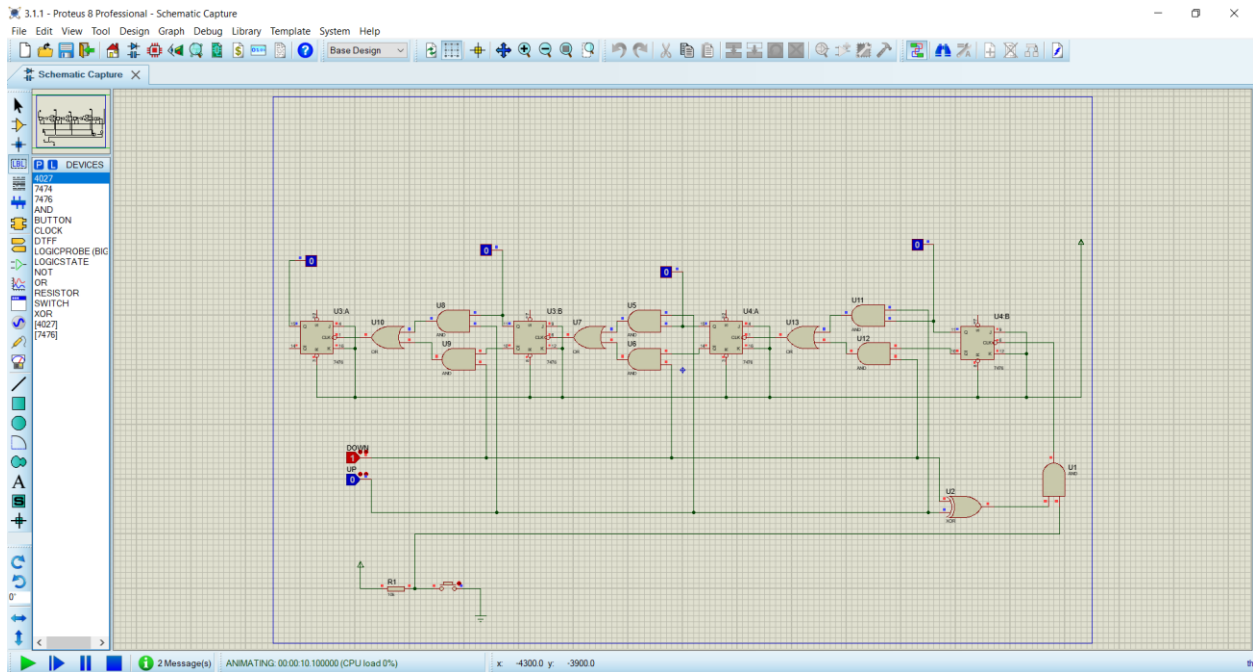
هدف در این آزمایش آشنایی و کار کردن با شمارنده ها چه سنکرون و چه آسنکرون است.

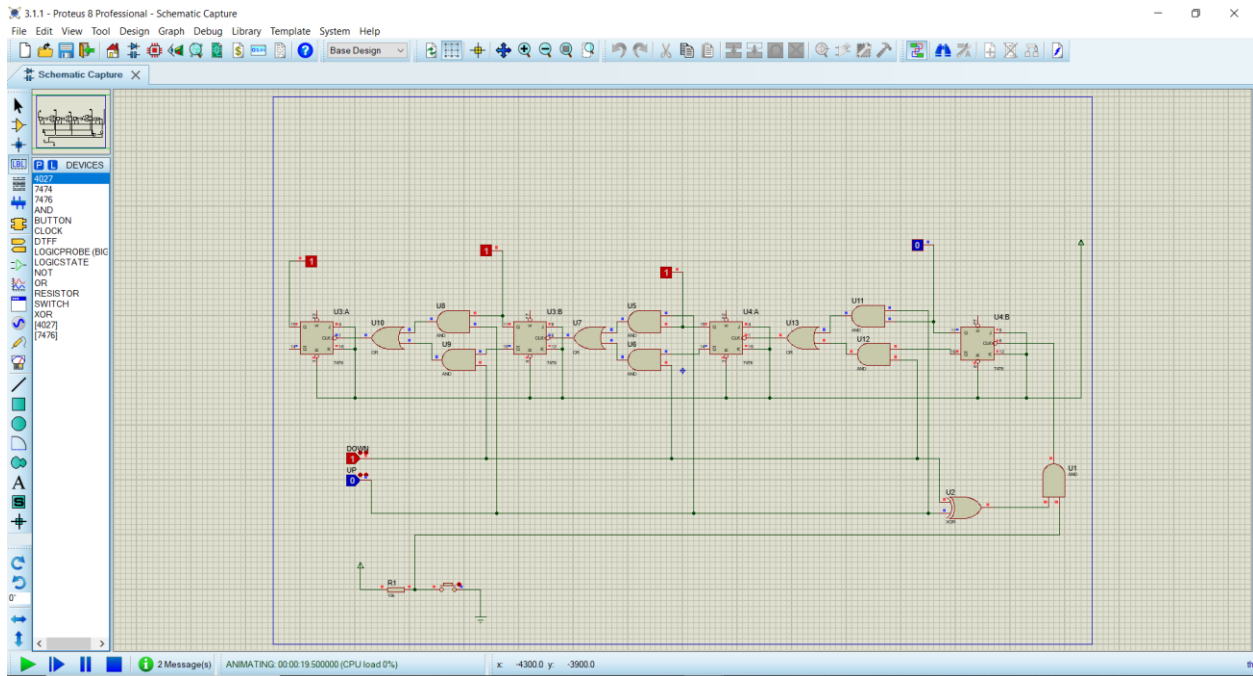
## آزمایش اول

### بخش اول :

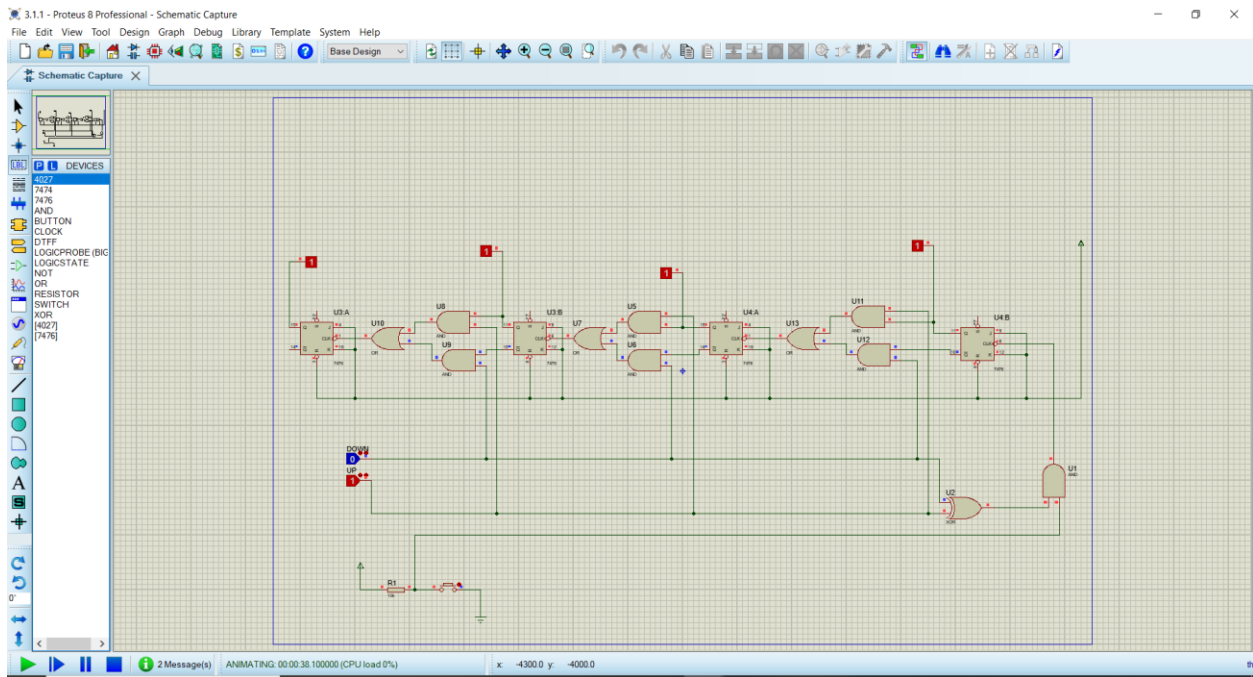
همانطور که در دستور کار بود در پروتئوس مدار مربوطه را می بندیم. تصاویر زیر نشان می دهند مدار درست کار می کند:

با  $down=1$  و زدن سه کلاک:





پس از آن یک کلاک و با  $up=1$ :

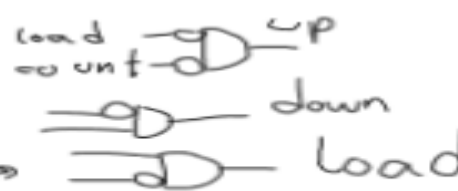


## بخش دوم

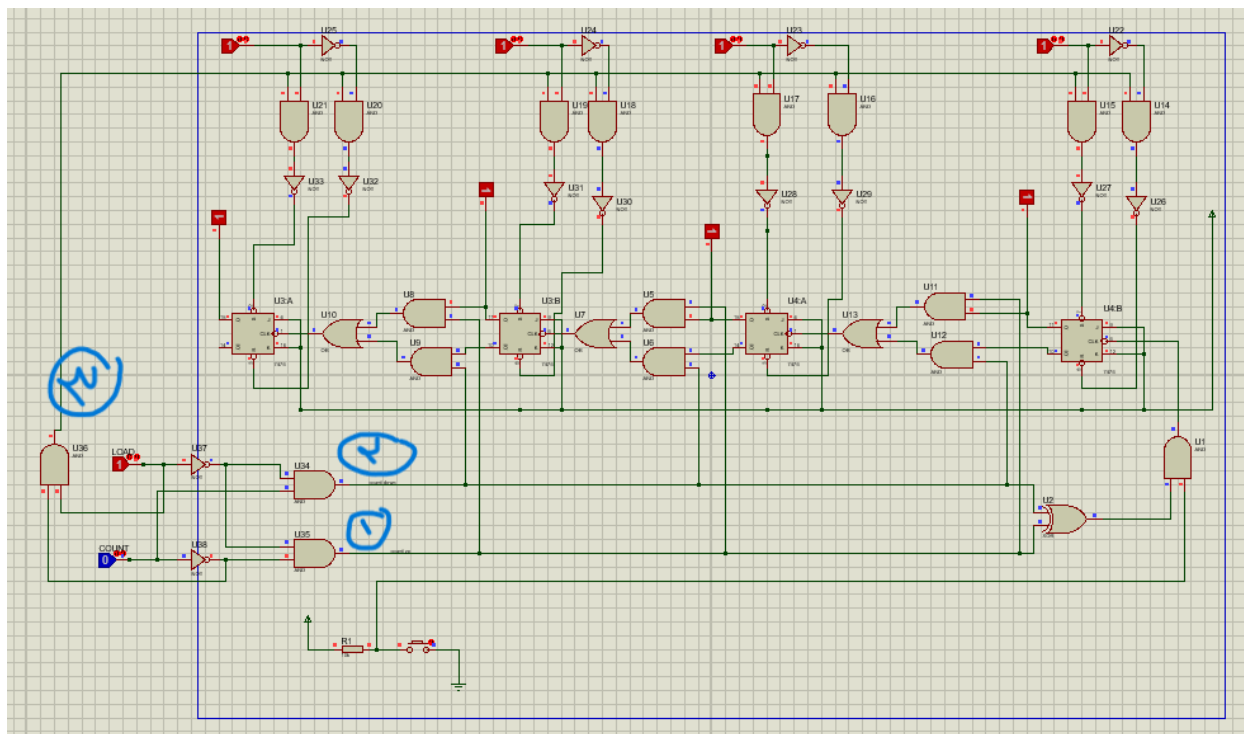
برای حداقل تغییرات حداقل امکان از امکانات بخش قبل استفاده می کنیم به این صورت که برای دادن ۴ ورودی مد نظر از set و reset استفاده کرده و همینطور بجای logic state ها up و down بخش دیگری (مطابق شکل) قرار می دهیم.

چگونگی فهمیدن ورودی ها از دستور مطابق جدول داده شده است:

load	count	function
0	0	count up
0	1	count down
1	0	parallel load
1	1	do nothing



که دو مورد اول را در مدار سابق داشتیم. برای مورد سوم مطابق شکل زیر عمل می کنیم تا هر وقت گیت سوم یک بود برای هر فلیپ فلاپ set یا reset رخ دهد (طبق ترتیب بالا در شکل شماره گذاری کردم):



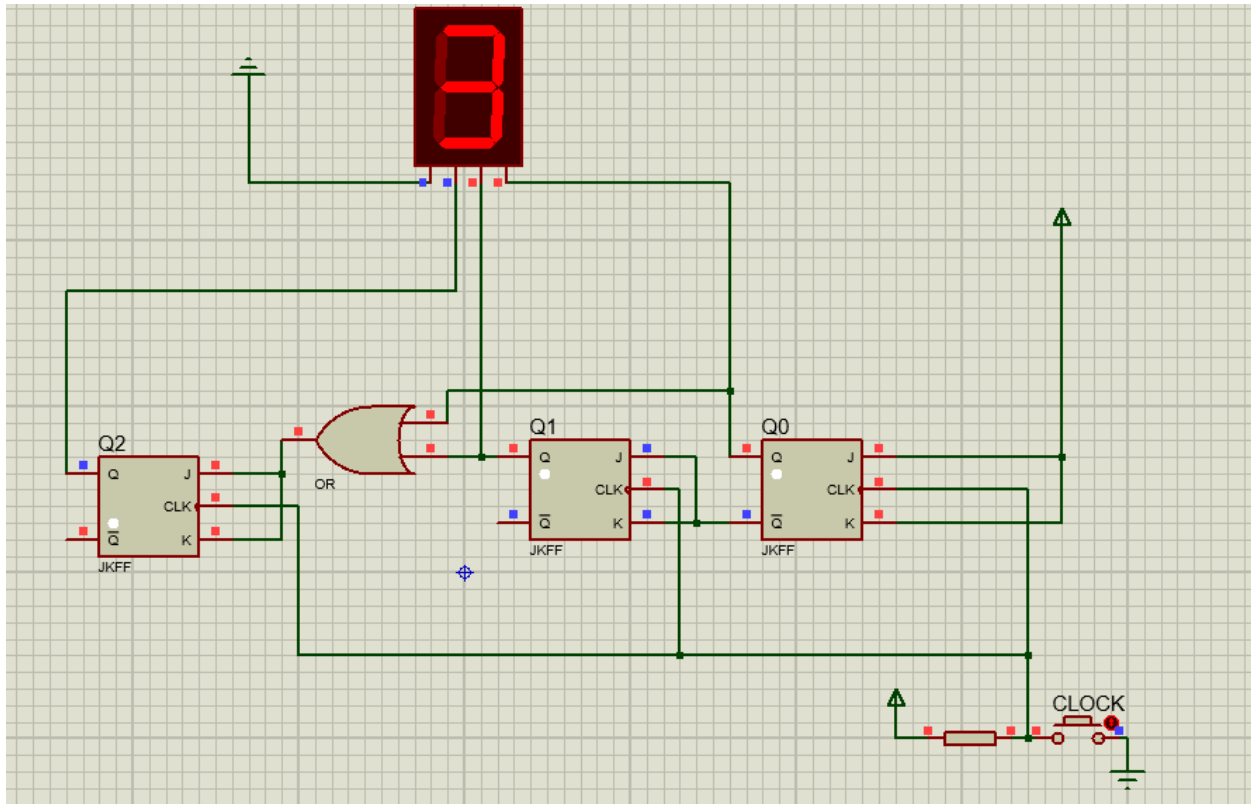
در مورد آخر هم هرگاه هر load و count ورودی ها ۱ باشند (حالت چهارم) خروجی هر سه گیت AND صفر می شود و اتفاقی رخ نمی دهد. فایل های fritzing همه آزمایش ها ضمیمه شده اند.

## آزمایش دوم

در این آزمایش یک شمارنده دو طرفه سنکرون با قدم (Step) های سه تایی با ۳ بیت طراحی میکنیم.

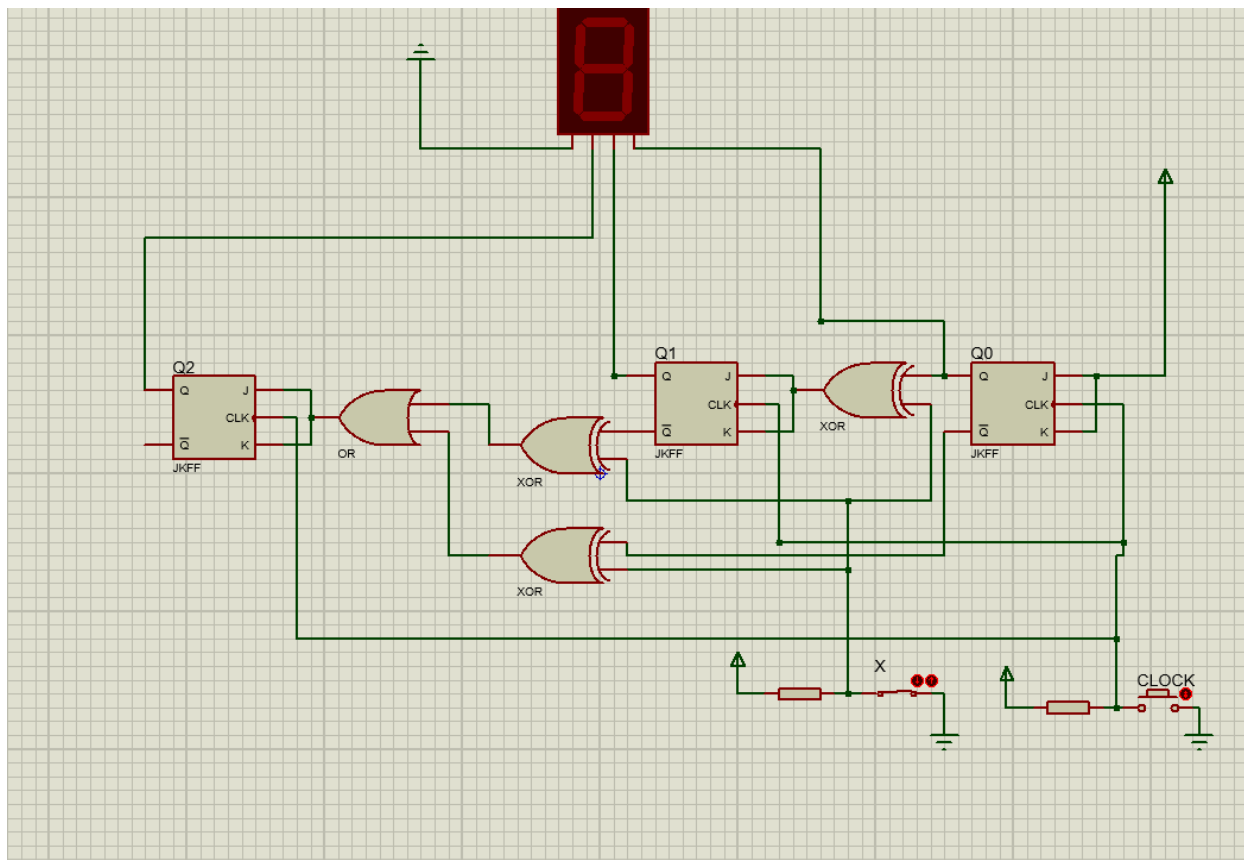
اول شمارنده بالارونده را طراحی میکنیم.

اگر قدم ها سه تایی باشد، در هر پالس رقم اول به هر حال تغییر میکند، از طرفی رقم دوم هم همیشه تغییر می کند مگر وقتی که رقم اول ۱ باشد که در آن صورت آن ۱ با ۱۱ جمع میشود و انگار ۱۰۰ اضافه می شود و رقم دوم ثابت می ماند، پس Toggle رقم دوم برابر معکوس رقم اول می باشد. تاگل رقم آخر هم با حالت بندی برابر  $OR(Q0, Q1)$  است.

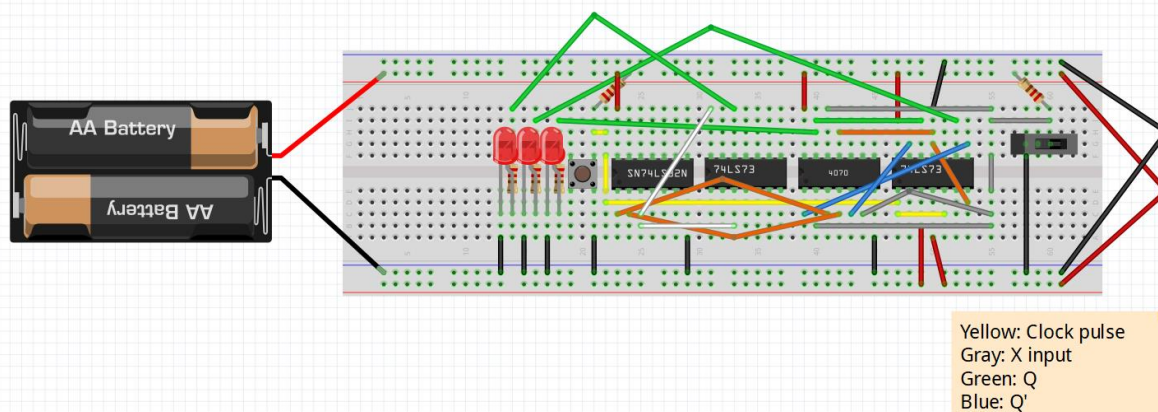


در قدم بعدی برای تغییر جهت شمارش، کافیهست در همه جا به فلیپ فلاپ بعدی به جای  $Q$ ،  $Q'$  بدهیم و برعکس، برای این هدف از XOR استفاده می کنیم که در یک حالت ورودی  $X$ ، دیگری را قرینه کند.

البته برای صرفه جویی در تعداد گیت به جای قرینه کردن  $X$  ورودی ها را برعکس کردیم.



همچنین طراحی مدار در فریتزینگ به شکل زیر است:

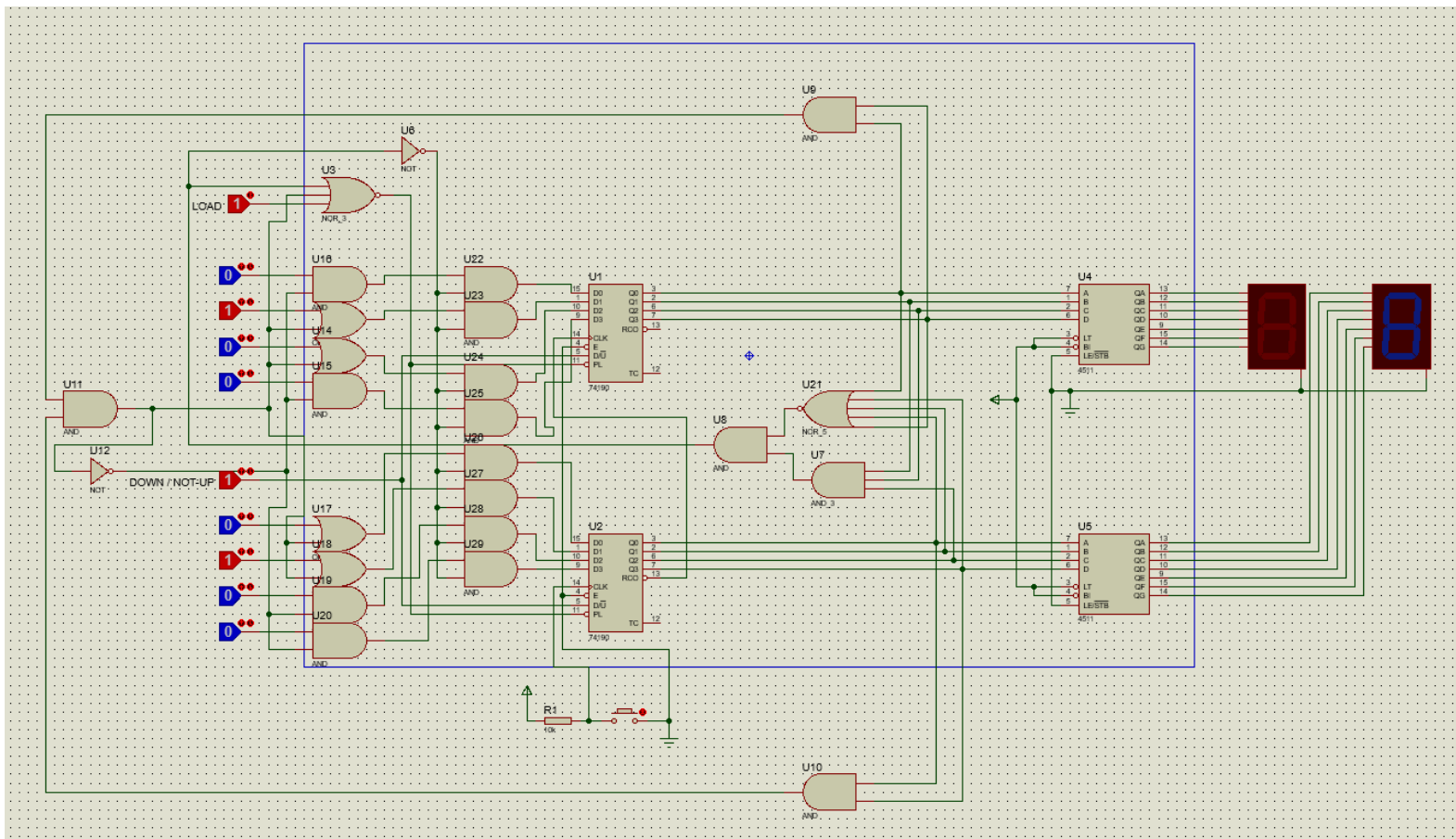


fritzing



## آزمایش سوم:

در این آزمایش، یک شمارنده BCD که از ۰ تا ۶۳ را به روش 7-segment نمایش می‌دهد می‌سازیم. این شمارنده، قابلیت لود کردن و شمارش رو به بالا و پایین را دارد. اگر دکمه لود در حالت یک باشد، ورودی‌ها لود می‌شوند و اگر دکمه شمارش رو به بالا/پایین صفر باشد، شمارش رو به بالا در غیر این صورت رو به پایین است.



برای کنترل ۰ شدن شمارنده ها هنگامی که در شمارش بالا به ۶۴ رسیدیم یا ۶۳ شدن شمارنده ها هنگامی که در شمارش پایین به ۹۹ رسیدیم، گیت های AND و OR در ورودی به شمارنده ها قرار داده شده اند. در عکس های پایین، دو کلاک متوالی از شمارش رو به پایین قرار داده شده است.

