

به نام خدا



آشنایی با محیط های شبیه سازی

استاد: دکتر شاهین حسابی

دانشگاه صنعتی شریف

تابستان 1401

امیرحسین علمدار

400105144

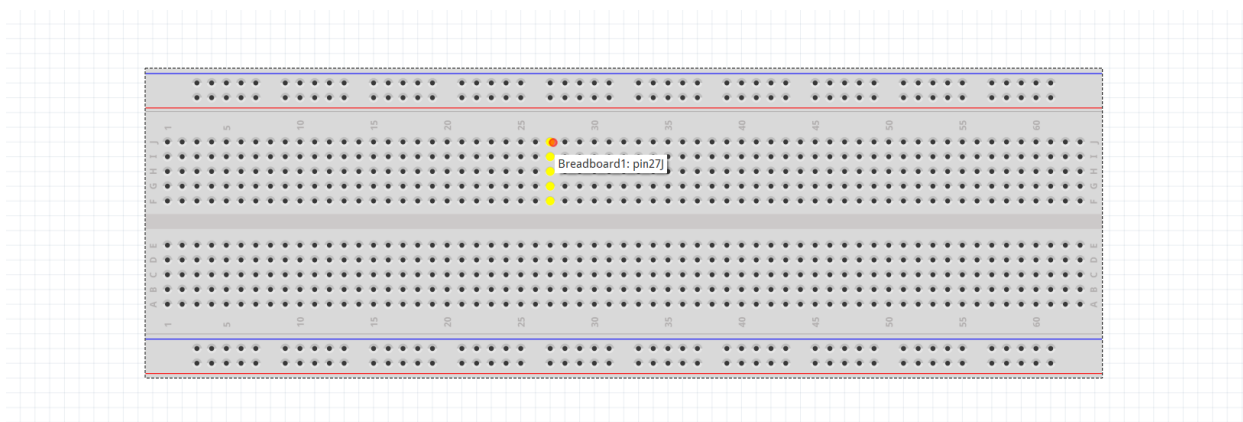
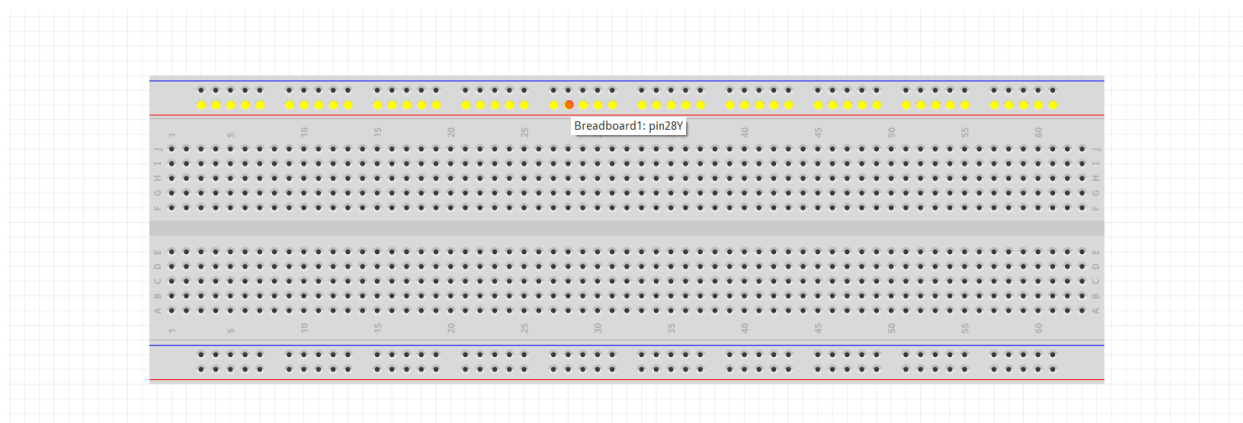
مقدمه

هدف از آزمایش آشنایی با نرم افزار های شبیه سازی مدار منطقی،

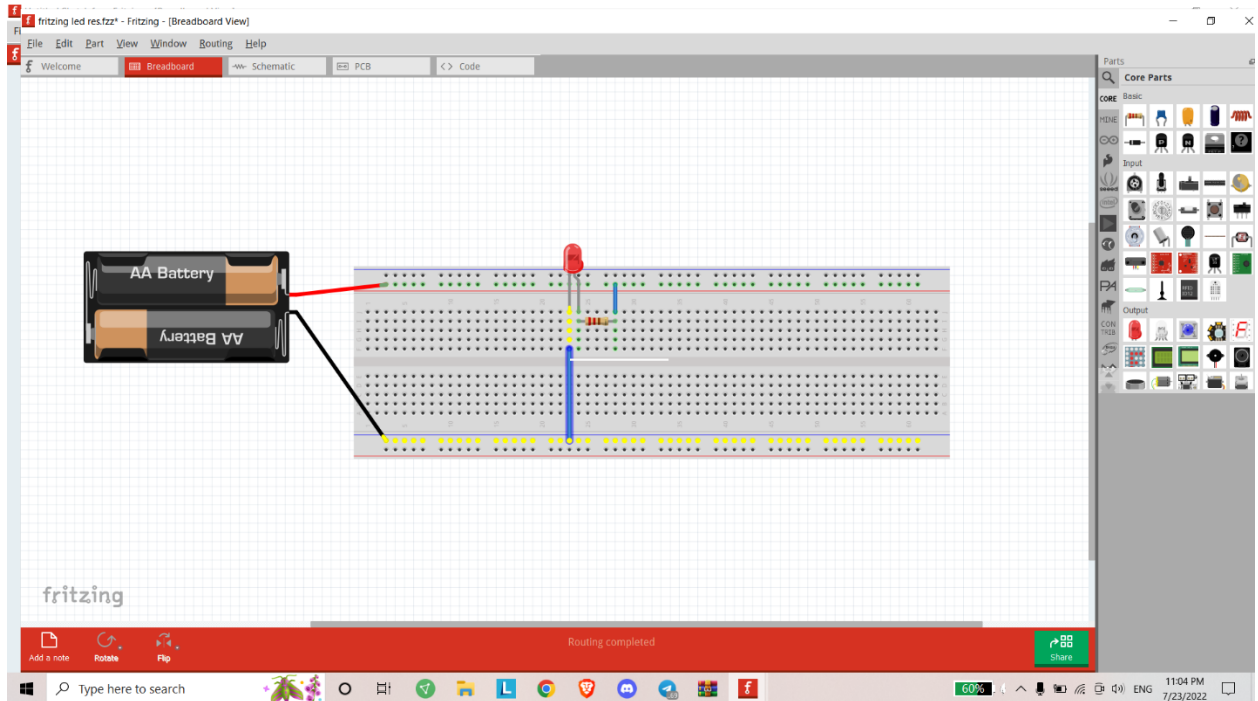
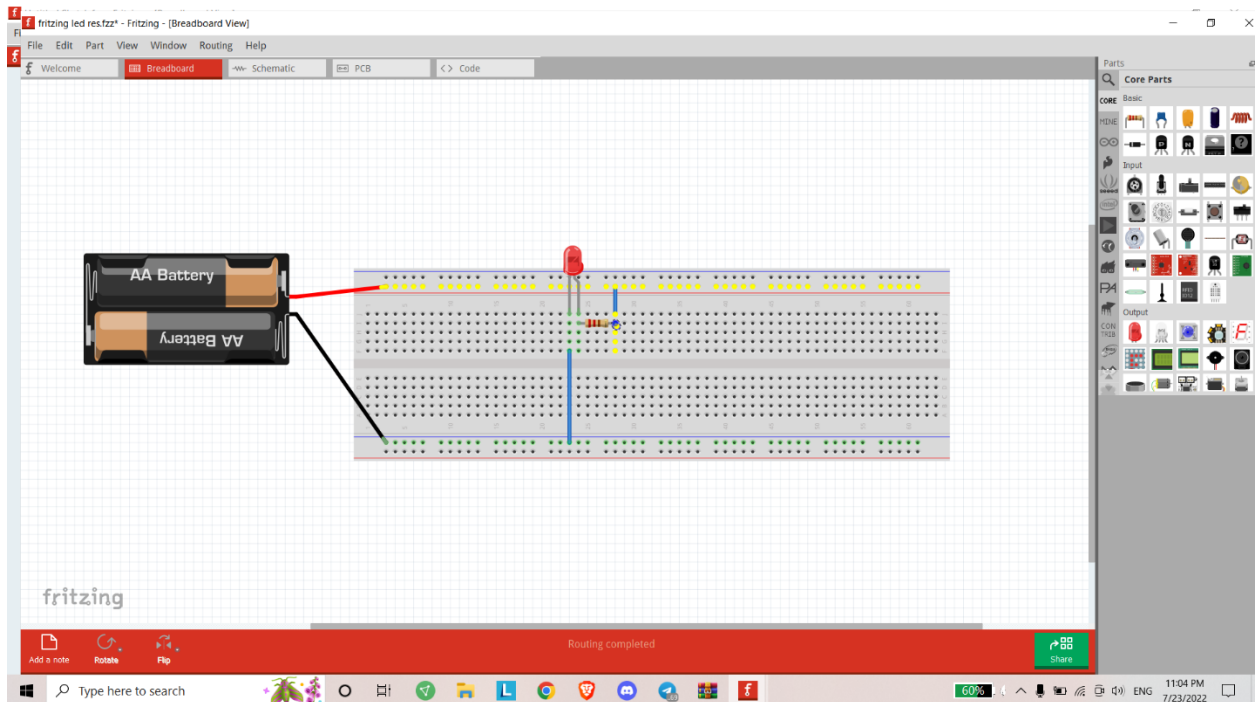
(Fritzing, Logisim, Proteus) است. این آزمایش در 3 بخش انجام می شود در بخش اول با طرز کار و اتصالات یک برد مورد آشنا می شویم. در بخش دوم نیز یک مدار ترکیبی ساده با Logisim می سازیم و در نهایت هم مدار ترکیبی پیچیده تری با Proteus رسم می کنیم.

بخش اول:

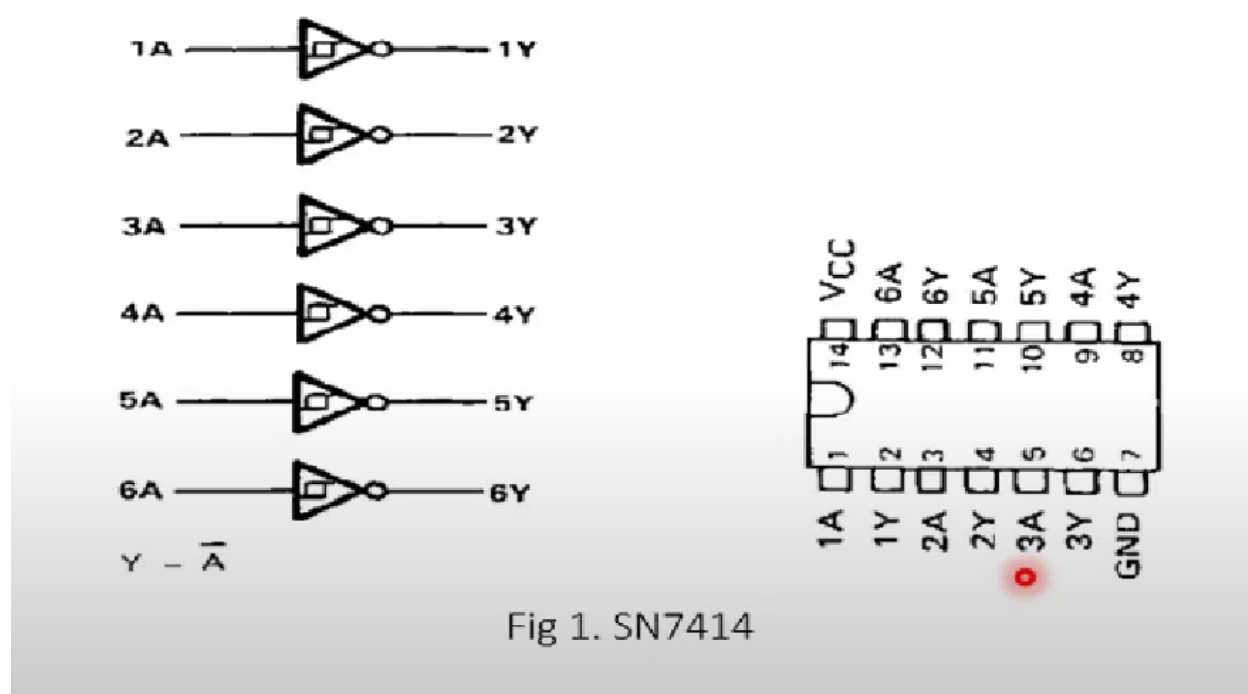
ابتدا اتصالات داخلی برد مورد را بررسی می کنیم:



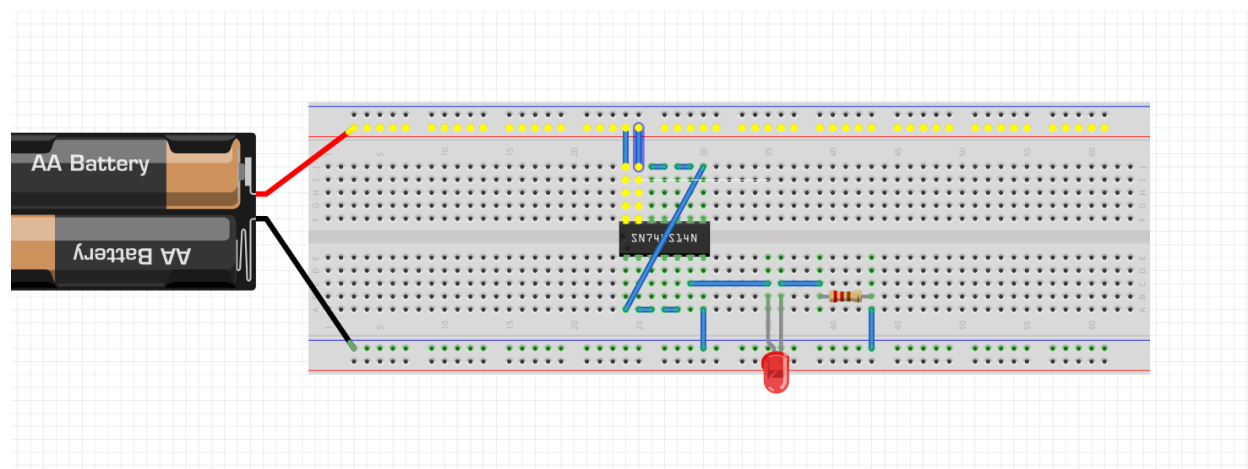
حال مدار ساده خواسته شده را با LED و مقاومت و باتری میسازیم و اتصالات نقاط مختلف را چک می کنیم:

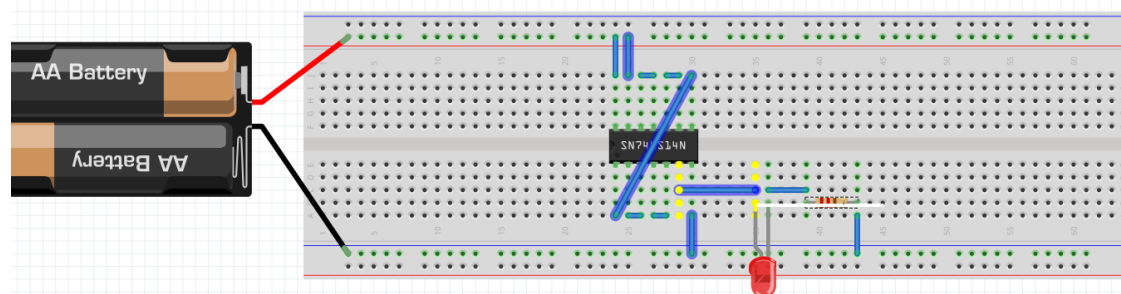
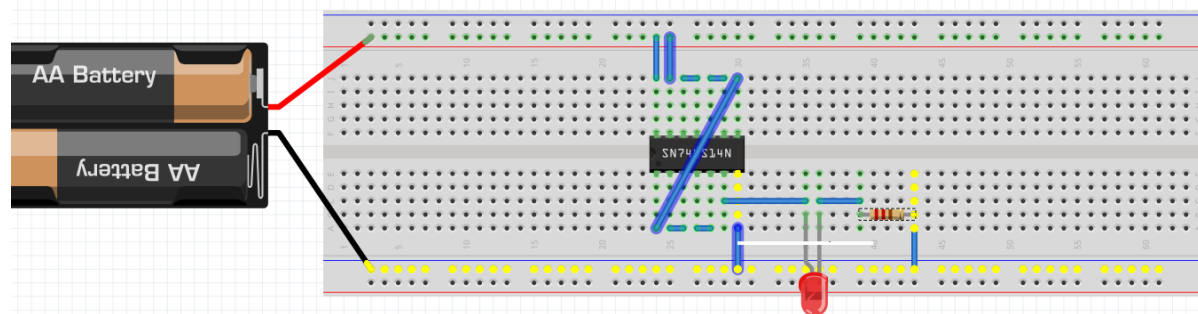
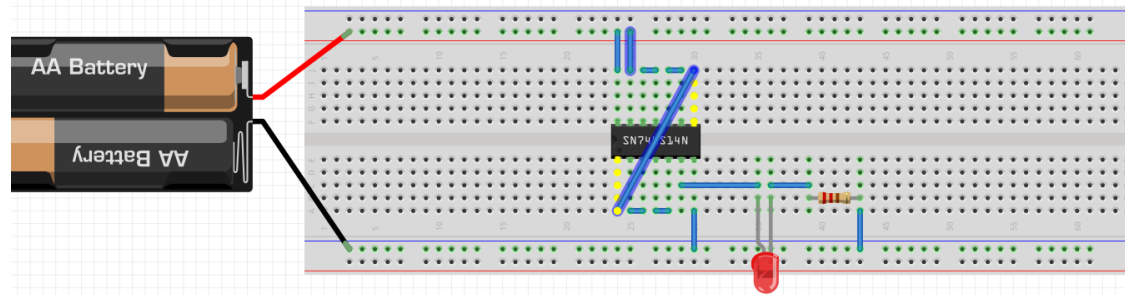


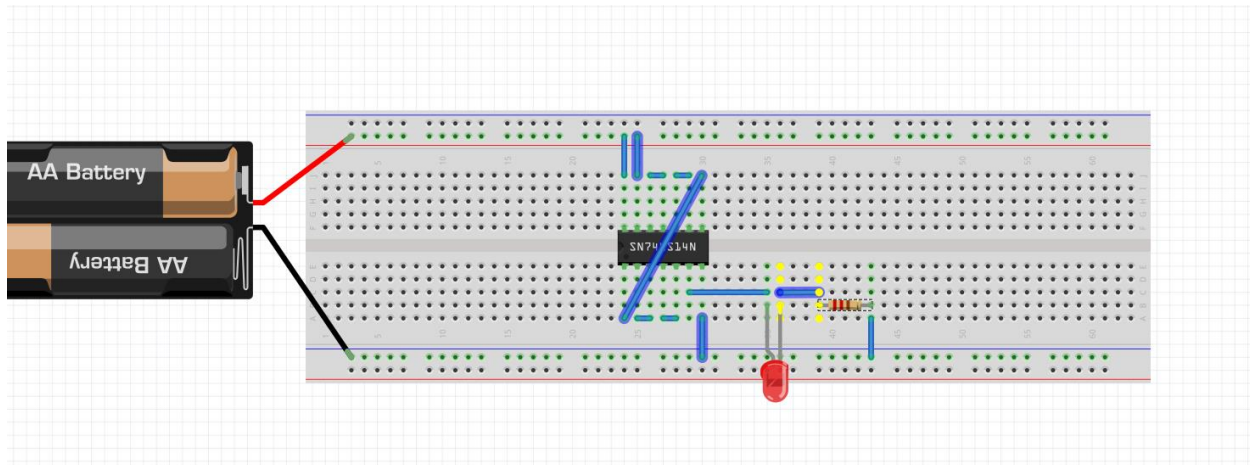
به سراغ مدار بعدی می رویم، برای ساخت این مدار از قطعه SN74LS14N استفاده می کنیم که جفت پایه های متوالی گیت نات را ارائه می دهند و به این ترتیب 6 گیت نات در اختیار ما قرار می دهد:



با استفاده از این اطلاعات مدار را می بندیم و اتصالات آن را چک می کنیم:





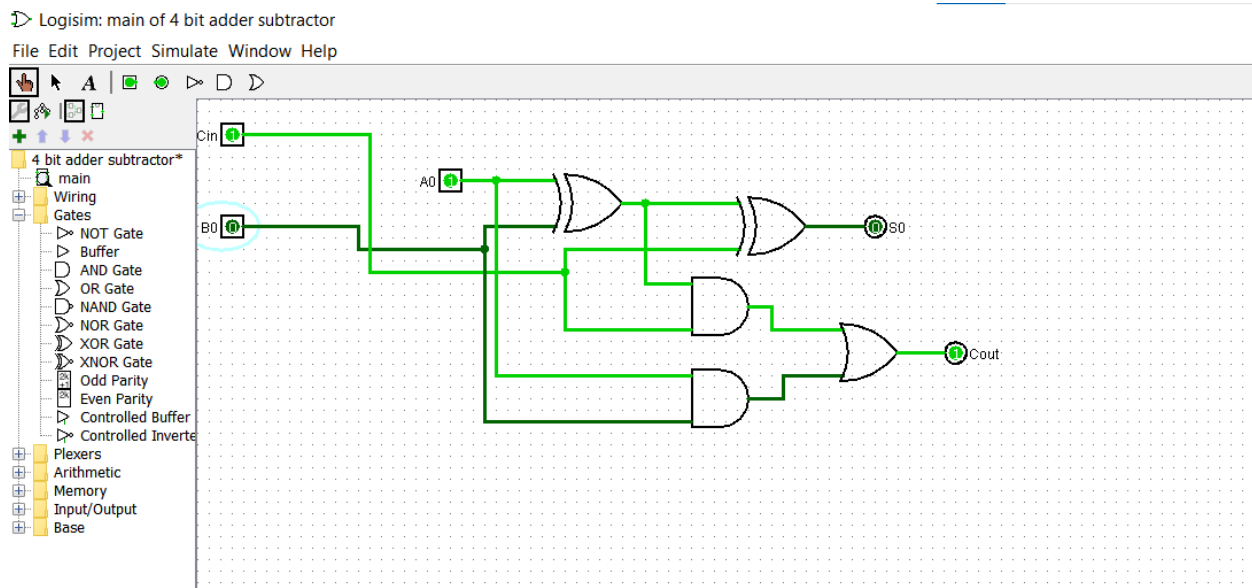


اگر در شرایط آزمایشگاه بودیم LED روشن می شد زیرا بعد 6 بار نات کردن ولتاژ اولیه که همان VCC بود را داشتیم که به قطب مثبت LED می رسد.

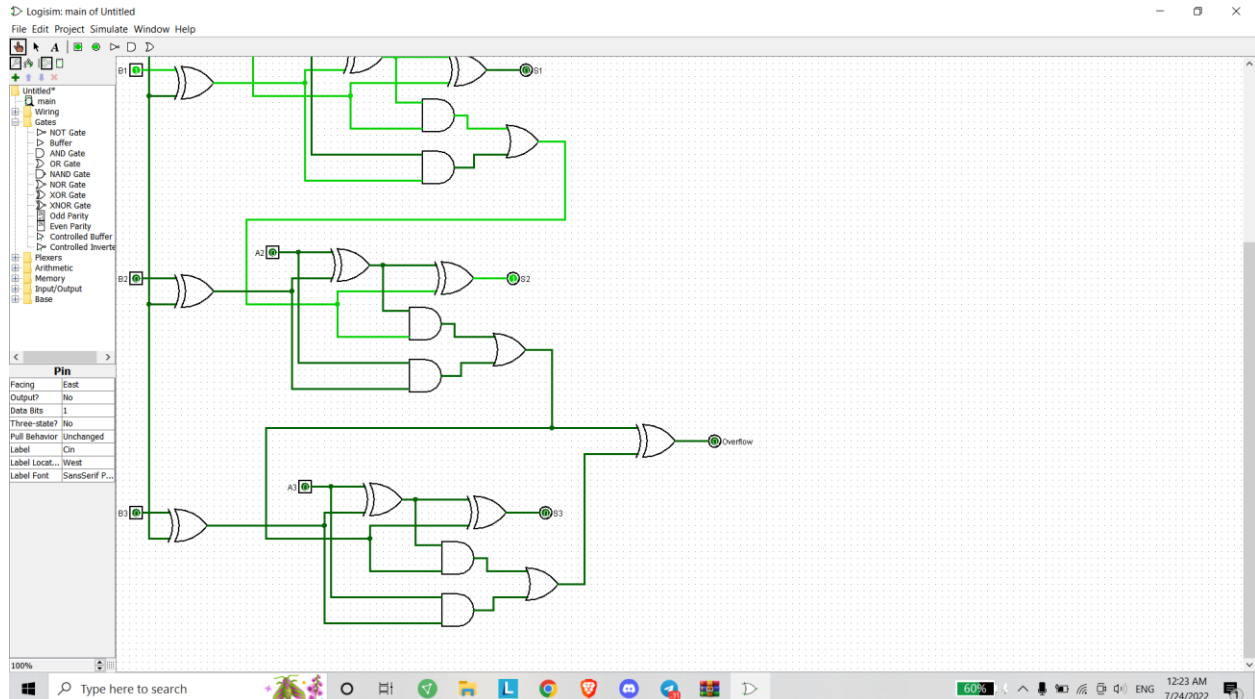
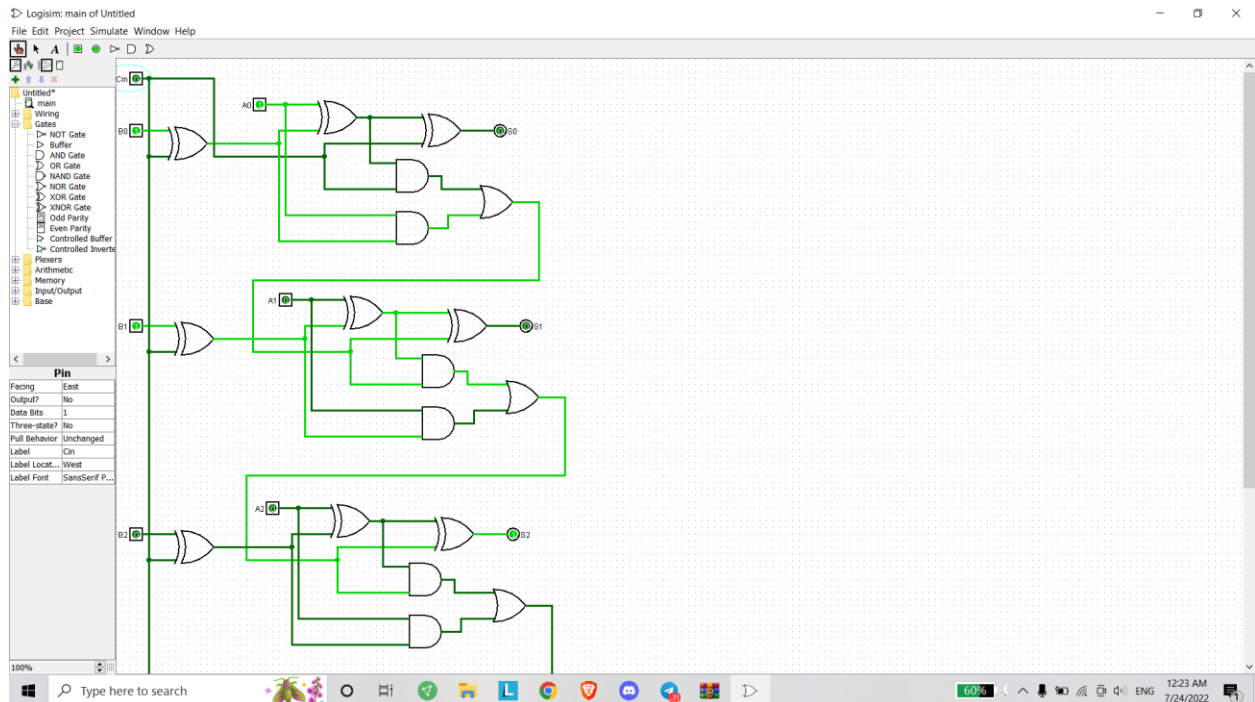
بخش دوم:

رسم مدار ها با Logisim:

ابتدا فول ادر را رسم می کنیم:



حال با 4 بار استفاده از فول ادر بالا یک جمع کننده/تفریق کننده 4 بیتی می سازیم که بسته به صفر یا یک بودن Cin ، جمع یا تفریق انجام می دهد(در حالت تفریق A-B انجام می شود):



همانطور که از درس مدار منطقی میدانیم، اگر C_{in} یک باشد، بیت های B نات میشوند و به جمع کننده مقدار یک اضافه می شود که عملاً داریم A را با مکمل دوم B جمع می کنیم که همان $A-B$ است. (با فرض اینکه این مطالب در مدار منطقی آموخته شده و هدف آزمایشگاه پیاده سازی عملی آن هاست از توضیحات دقیق تئوری مانند جدول کارنو و نشان دادن روابط صرف نظر می کنیم)

بخش سوم:

همانطور که می دانیم در مدار Carry-Look-Ahead روابط زیر برقرار اند:

$$\begin{aligned} P_i &= A_i \oplus B_i \\ G_i &= A_i B_i \\ S_i &= P_i \oplus C_i \\ C_{i+1} &= G_i + P_i C_i \end{aligned}$$

پس ما ابتدا P و G ها را از ورودی ها بدست می آوریم و سپس با استفاده از آن ها ابتدا کری ها را به ساده ترین صورت می نویسیم طوری که هر کری تابعی از کری های پیشین نمایش داده نشود و صرفاً تابعی از کری صفر و P و G ها باشد. بدین صورت دیگر لازم نیست منتظر کری های قبلی برای بدست آوردن یک کری باشیم و تاخیر مدار کم می شود. حال که هر کری را بدست آوردیم آن را با P متناظر ایکس اور می کنیم تا حاصل جمع بدست بیاید:

با بستن مدار ها با برنامه های مورد نظر آشنا شدیم و دریافتیم برنامه **Fritzing** بیشتر برای آشنایی با قطعات و شرایط آزمایشگاه است و قابل شبیه سازی نیست. همچنین **Logisim** در لحظه شبیه سازی می کند و ازین نظر که موقع بستن مدار از شرایط آن آگاهیم می تواند کمک کند. در نرم افزار **Proteus** نیز با درخواست شبیه سازی، شبیه سازی انجام می شود همچنین می توان لایبرری هایی برای آن نصب کرد تا قطعات بیشتری داشته باشیم و همچنین این برنامه رابط کاربری بهتری دارد.