بسمه تعالی

گزارش کار اول آزمایشگاه معماری کامپیوتر

جمع کننده ده دهی

استاد: دکتر سربازی

دستیار آموزشی: سرکار خانم غیبی

نویسندگان

فرزام کوهی رونقی ۴۰۱۱۰۶۴۰۳

مریم شیران ۴۰۰۱۰۹۴۴۶

ثنا بابایان ونستان ۴۰۱۱۰۵۶۸۹

دانشگاه صنعتی شریف

تابستان ۱۴۰۳

فهرست مطالب

۱ آزمایش اول: جمع کننده ده دهی

۱-۱ مقدمه و هدف

هدف از این آزمایش، طراحی یک جمع کننده سه رقمی BCD است. به این معنی که این جمع کننده دو عدد سه رقمی در مبنای ده ورودی گرفته و در خروجی حاصل جمع آن ها را نمایش خواهد داد. ضمن این که به عنوان آزمایش اول، یک هدف دیگر آن هم آشنایی با کار با نرم افزار Proteus است.

۱-۲ تجزیه و تحلیل تئوری آزمایش

در سیستم BCD برای هر رقم در مبنای ۱۰ به اندازه ٤ بیت در مبنای ۲ در نظر گرفته می شود به طور مثال عدد $(۹۵۲)_{۱۰}$ به صورت $(۱۰۰۱\ ۰۱۰۱\ ۰۰۱۰)_{BCD}$.برای جمع زمانی که حاصل از ۱۰ بیش تر یا مساوی باشد، نیازمند بررسی بیشتر هستیم تا بتوانیم carry آن را به طور جدا خروجی بدهیم و حاصل یکان عدد حاصل بشود. برای این کار فرض کنید عددی که از Full-Adder ۴Bit خارج شده است، به صورت C،a۳a۲a۱a۰ باشد که C مقدار Carry خروجی مبنای دو و سه رقم بعدی هم خود عدد خروجی هستند .اعداد خروجی می توانند یکی از سه حالت زیر باشند :

۱. عدد حاصل ۱۰ یا ۱۱ باشد :

   برای بدست آوردن یکان در این حالت هم میتوان مانند قبل از اضافه کردن ٦ استفاده کرد. برای شناسایی وجود carry هم می توان از این نکته استفاده کرد که
   a۳ AND a۱ = ۱.

۲. عدد حاصل جمع بین ۱۲ تا ۱۵ باشد :

   در این حالت با اضافه کردن ٦ واحد به خود خروجی می توان یکان را بدست آورد (مثلا ۰۰۱۰ → (+۰۱۱۰)۱۱۰۰ ⟵ ۱۲) و از سوی دیگر برای این که وجود carry در چنین حالتی را شناسایی کنیم کافیست دقت کنیم که در همه این اعداد
   a۳ AND a۲ = ۱ است.

۳. عدد حاصل جمع بین ۱٦ تا ۱۸ باشد :

   مقدار a۳a۲a۱a۰ ٦ واحد کمتر از یکان مد نظر خواهد بود مثلا برای ۱٦ یکان مورد انتظار ۰۱۱۰ خروجی واقعی پس از جمع ۰۰۰۰۰

جمع دو رقم با هم تا زمانی که حاصل از ۹ فراتر نرود، یک جمع کننده Full-Adder ۴Bit معمول هم کافیست.(۱ شدن C به این معنا است که carry داریم.) حال می توانیم یک رابطه کلی برای carry ارایه دهیم :

$$C_{BCD} = C + (a۳ \text{ AND } a۲) + (a۳ \text{ AND } a۱)$$

۱-۳ شرح دستگاه ها و وسایل مورد استفاده

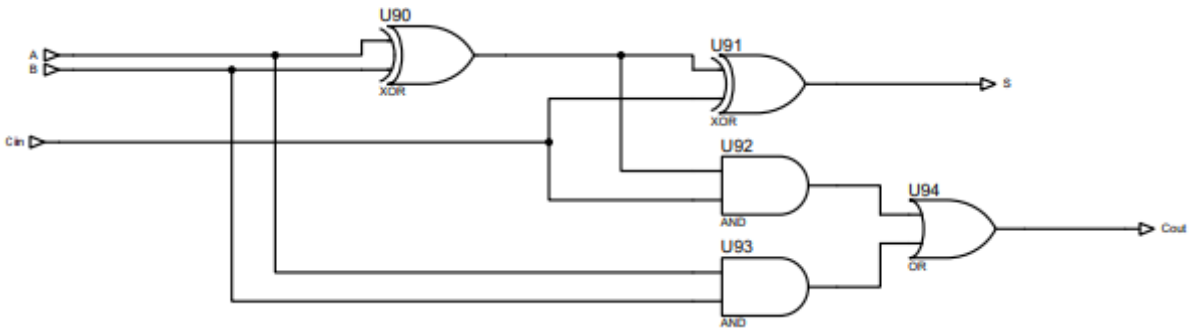
برای شبیه سازی تیوری نرم افزار Proteus استفاده میشود.

برای شبیه سازی عملی از برد بورد، سیم ،تراشه ی ۷۴۰۸ Quad 2 input AND Gate، ۷۴LS۸۳ Quad Exclusive OR Gate، Quad 2-Input OR Gate ۷۴۳۲، 4-bit Binary Full Adder ۷۴۸۶ ،منبع تغذیه ،مقاومت و لامپ LED استفاده میشود.

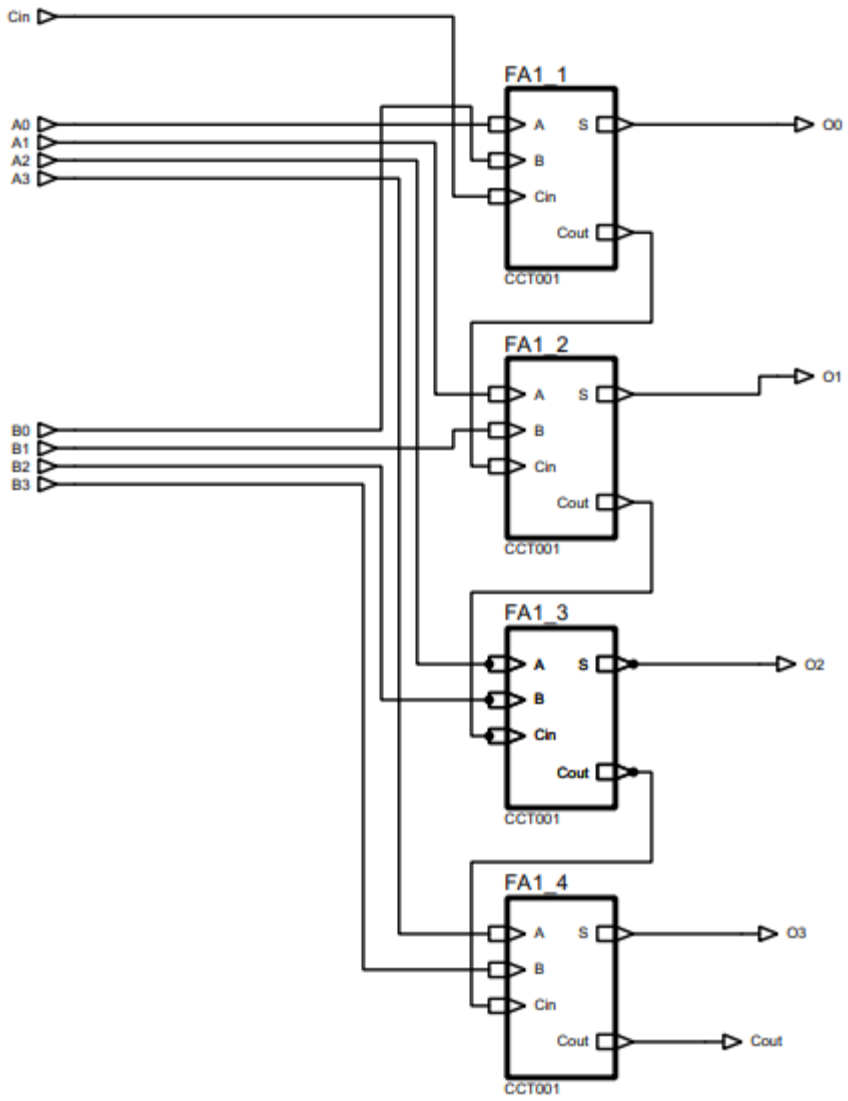
۱-۴ شرح آزمایش

۱-۴-۱ در پروتیوس

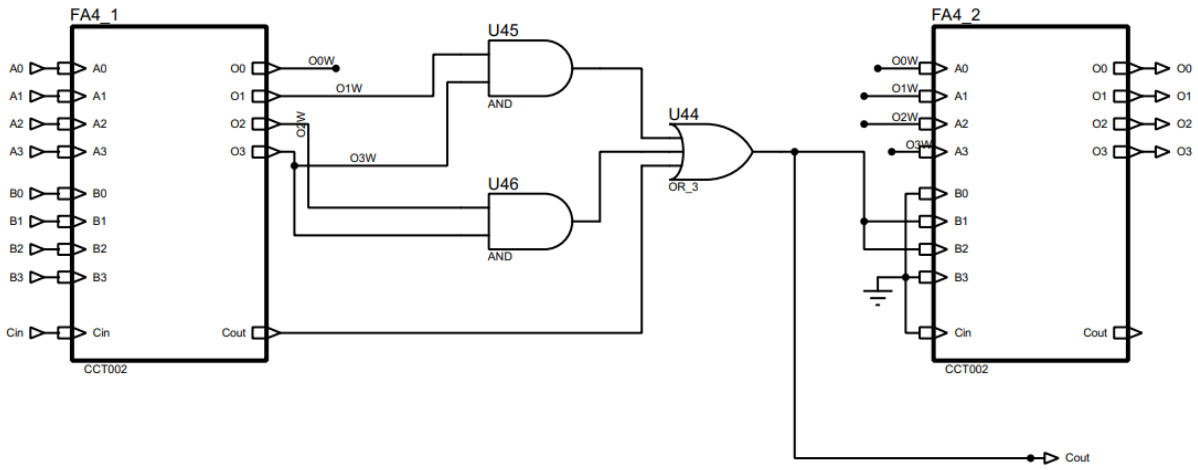
Full Adder ای که برای یک بیت از آن استفاده می کنیم در این طراحی به شکل زیر است که A, Cin, B ورودی تک بیتی و Cout , s خروجی تک بیتی ما می باشند :



حال برای اینکه بتوانیم دو عدد سه بیتی را با هم جمع کنیم باید یک Full Adder ٤ بیتی طراحی کنیم که در اینجا از طراحی Ripple Carry Adder استفاده می کنیم که یعنی carry ورودی هر واحد تک بیتی به از carry خروجی قبلی استفاده می شود :

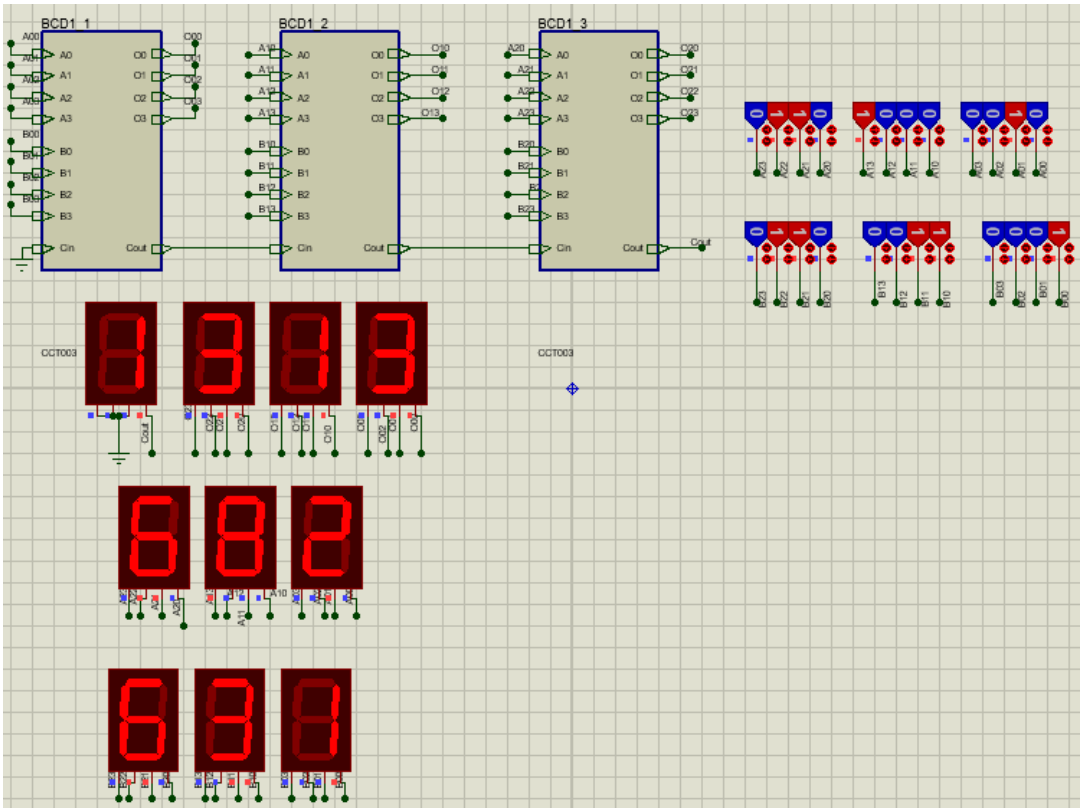


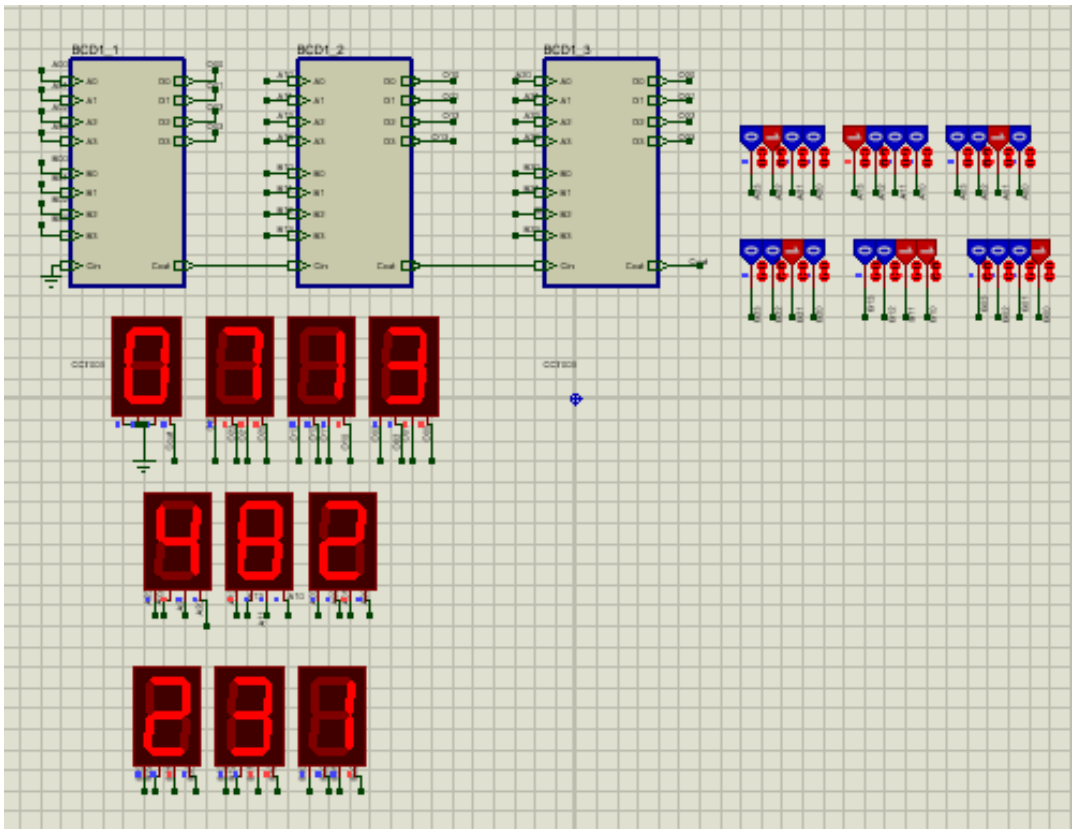
برای تبدیل خروجی جمع کننده اول به خروجی BCD مد نظر، مشخص شد که باید در صورتی که مقدار آن بیش تر مساوی ۱۰ بود، آن را با ٦ جمع کنیم. برای این کار می توانیم یک تمام جمع کننده چهاربیتی دیگر قرار بدهیم که در صورتی که carry وجود داشت، مقدار خروجی با ٦ و در غیر این صورت با ۰ جمع شود. برای این کار هم می توانیم مقدار C_{BCD} را به عنوان بیت اول و دوم عددی که باید با حاصل قبلی جم شود، ورودی داده و بیت اول و چهارم را هم صفر بگذاریم :



برای نمایش خروجی ها و دادن ورودی ها از ده عدد 7-segment استفاده نمودیم بدین ترتیب که سه تا برای ورودی اول، سه تا برای ورودی دومو چهارتا برای خروجی.(رقم آخر خروجی از طریق cout آخرین جمع کننده BCD تک بیتی حاصل می شود.)

برای ورودی دادن دو ردیف LogicState قرار گرفته است.ردیف اول ربای خروجی و ردیف دوم برای ورودی اول و ردیف سوم برای ورودی دوم است. در هر ردیف ۱۲ عدد LogicState قرار گرفته که از سه قسمت ٤ تایی تشکیل شده اند .چهارتای سمت چپ برای صدگان، چهارتای وسط برای دهگان و چهارتای سمت راست برای یکان هستند .هر چهار بیت بیانگر یک رقم هم به همان شکل ریاضیاتی هستند. یعنی سمت چپ ترین بیت بیانگر MSB و سمت راست ترین آن ها هم LSB است. در شکل زیر که تصویر نهایی را نشان می دهد ورودی بالا بیانگر عدد ٤٨٢ و ورودی پایین بیانگر ٢٣١ است.٤٨٢ + ٢٣١ = ٧١٣ و یا ٦٨٢ + ٦٣١ = ١٣١٣.


BCD1_1
BCD1_2
BCD1_3
CCT003
CCT003

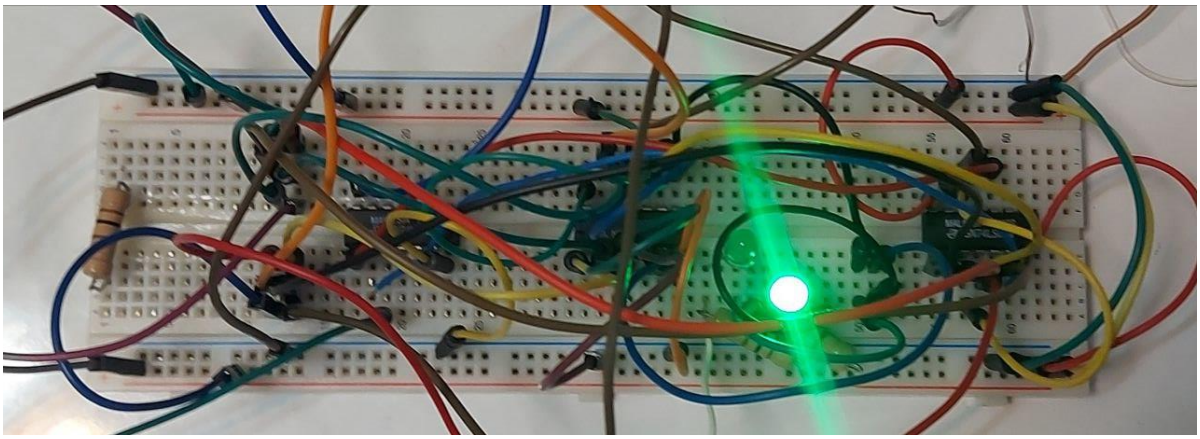

BCD1_1
BCD1_2
BCD1_3

۱-۴-۲ به صورت عملی

در کلاس با توجه به اینکه مدار ها را تا جای ممکن از پایه می بستیم؛ تنها برای یک رقم انجام دادیم.

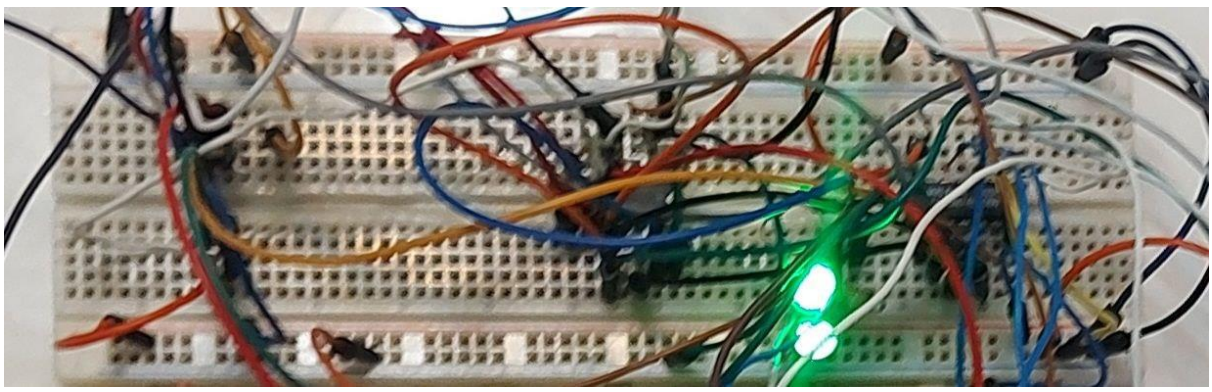
برای جمع دو رقم در مبنای ۱۰ نیاز داریم ۴ بیت باینری را باهم جمع کنیم. در زیر جمع کننده ۲ بیت سمت راست را با استفاده از تراشه های AND , OR , XOR پیاده سازی کرده ایم. برای مشخص کردن خروجی مدار برای هر خروجی یک LED گذاشتیم که برای جلوگیری از سوختن آن، برای هر یک از یک مقاومت نیز استفاده کردیم.

سپس VCCو GND برد بورد را به منبع تغذیه متصل نمودیم و در ادامه آن ها را به VCCو GND برد بورد های بعدی وصل نمودیم.



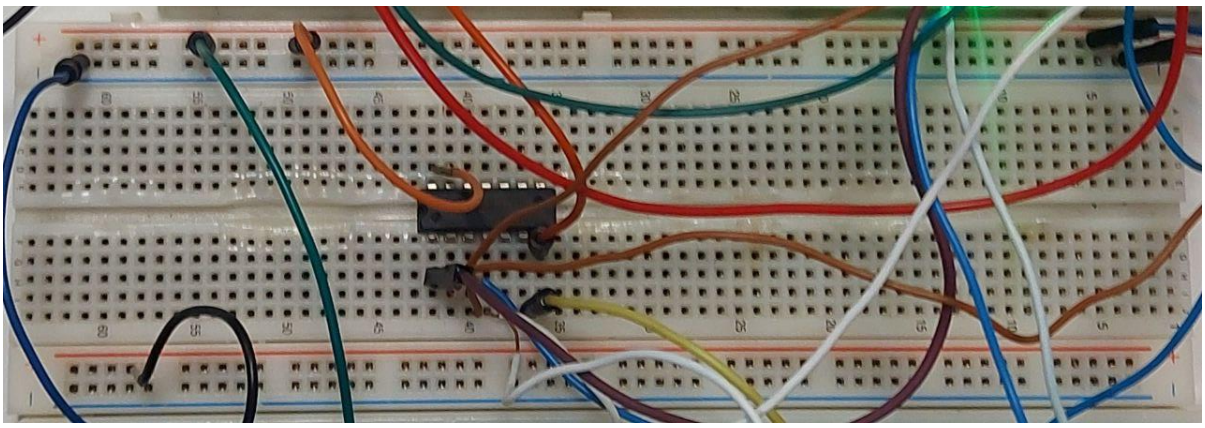
سمت راست تراشه ۷۴۳۲ وسط تراشه ۷۴۸۶ سمت چپ تراشه ۷۴۰۸

در زیر برای دو بیت دیگر سمت چپ نیز مطابق بالا جمع کننده ۲ بینی با استفاده از تراشه های AND , OR , XOR پیاده سازی کرده ایم. همچنین دقت کنید که Carry out برد بورد قبلی به عنوان Carry in در این برد بورد وارد میشود.



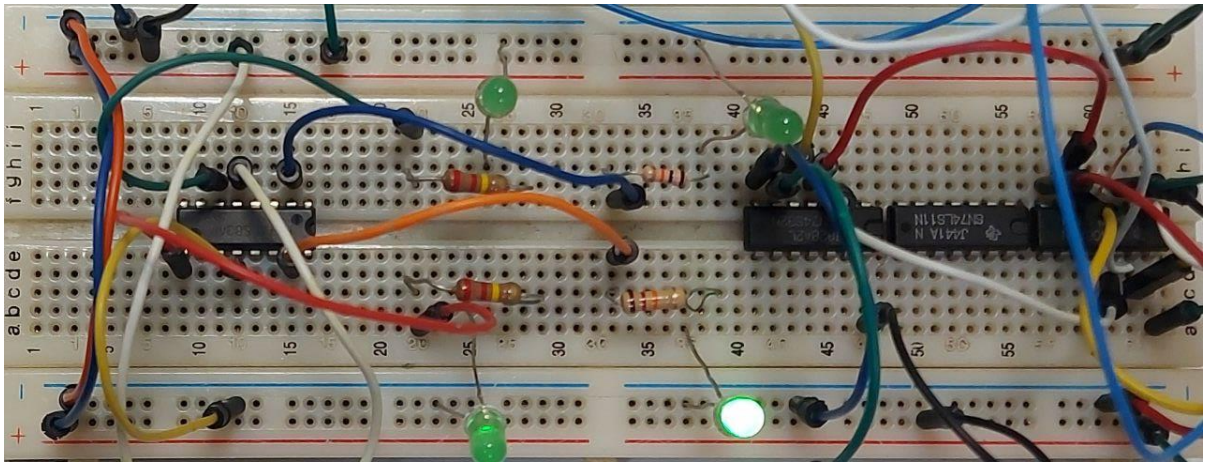
سمت راست تراشه ۷۴۳۲ وسط تراشه ۷۴۸۶ سمت چپ تراشه ۷۴۰۸

سپس در زیر Carry out جمع ۴ بیتی را پیاده سازی کردیم.



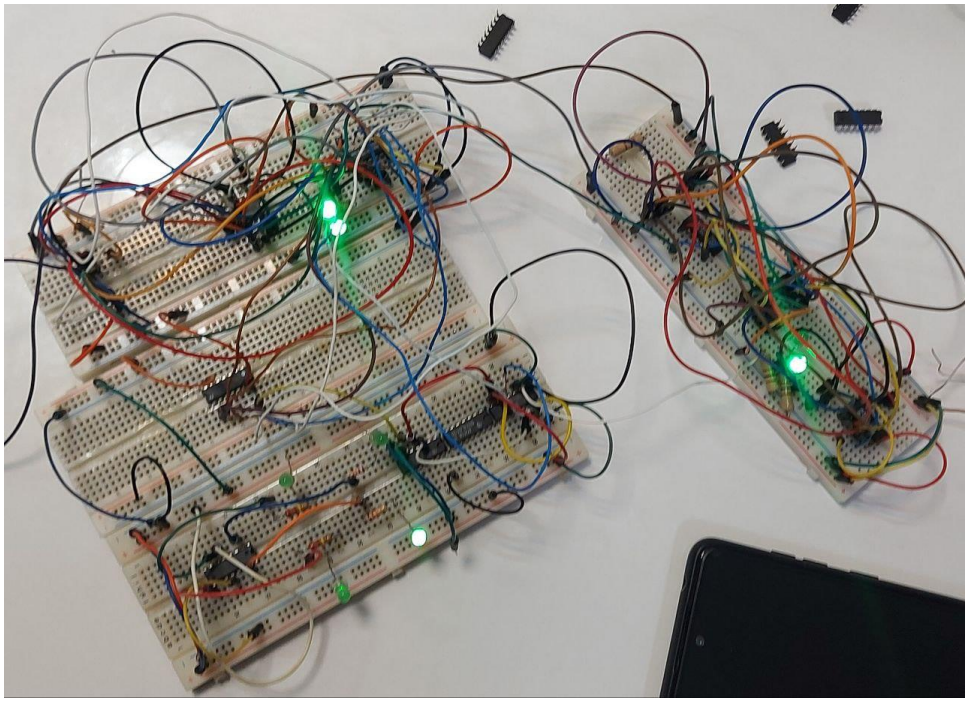
تراشه ۷۴۳۲

حال در نظر داشته باشید که آنچه که ما تا کنون ساختیم یک جمع کننده ۴ بیتی بود. اما برای جمع دو رقم ده دهی ما نیاز داریم که یک جمع کننده ۴ بیتی دیگر برای جمع با عدد ۶، هنگامی که حاصل جمع از ۹ بیشتر میشود داشته باشیم. برای این جمع کننده ی دوم از تراشه فول ادرر ۴ بیتی آماده استفاده کردیم.
و نیز برای دیدن خروجی های نهایی مدار مقاومت و LED گذاشتیم.



به ترتیب از سمت راست:
تراشه ۷۴۸۶ ، تراشه ۷۴۰۸ و تراشه ۷۴۳۲ و تراشه ۷۴۸۳

در پایان مدار نهایی به قسم زیر شد:



۵-۱ منابع و مراجع

- وبسایت گیکزفورگیکز
- وبسایت یوتیوب
- وبسایت ویکی‌پدیا