دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر
گزارش کار اول درس آزمایشگاه معماری

عنوان:

طراحی جمع کننده دهدهی

BCD Adder Design

نگارش

کیان قاسمی ۴۰۱۱۰۲۲۶۴
آیین پوست فروشان ۴۰۱۱۰۵۷۴۲
دیبا هادی اسفنگره ۴۰۱۱۱۰۲۴۵

استاد

دکتر حمید سربازی آزاد

دستیار آموزشی

مهندس عطیه غیبی فطرت

تیر ۱۴۰۳

فهرست مطالب

فهرست تصاویر

۱ مقدمه

در این آزمایش قصد داریم با استفاده از گیت های ابتدایی یک جمع کننده دهدهی بسازیم، در انتها مدار نهایی یک مدار ترکیبی است که دو عدد سه رقمی در مبنای ده دریافت می کند و جمع آن ها را در مبنای ده خروجی می دهد.
برای سادگی در پیاده سازی فیزیکی در آزمایشگاه، جمع کننده یک رقمی BCD با استفاده از IC های جمع کننده ۴ بیتی ساخته شد، البته به صورت جداگانه نیز یک جمع کننده ۴ بیتی با استفاده از گیت های AND, OR و XOR ساختیم. سلسله مراتب کلی طراحی این مدار به صورت زیر است:

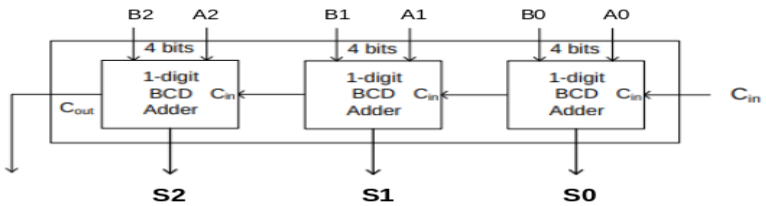
۱. طراحی بلوک تمام جمع کننده یک بیتی
۲. طراحی جمع کننده تک رقمی با استفاده از $full\ adder$ ساخته شده در مرحله قبل
۳. ساخت جمع کننده دهدهی سه رقمی با استفاده از جمع کننده های تک رقمی مرحله قبل
۴. ساخت چراغ هشدار دهنده برای حالتی که رقم ورودی بیش تر از ۹ باشد
۵. آزمایش کردن عملکرد درستی مدار با ورودی های مختلف

در نهایت بلوک دیاگرام نهایی برای شبیه سازی مدار و قطعات استفاده شده در پیاده سازی فیزیکی به صورت زیر می باشد:

۱.۱ قطعات لازم

۱. گیت xor دو ورودی (تراشه 7486)
۲. گیت and دو ورودی (تراشه 7408)
۳. گیت or دو ورودی (تراشه 7432)
۴. $4-bit\ FA$ (تراشه 7483)
۵. بردبورد و قطعات سیم

۲.۱ بلوک دیاگرام نهایی



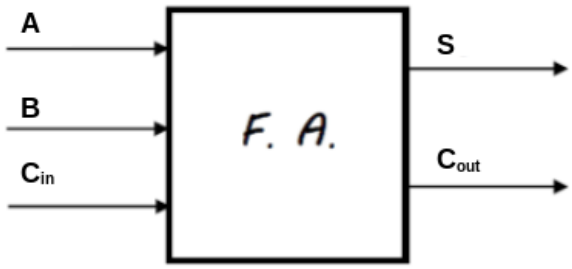
شکل ۱: بلوک دیاگرام نهایی

۲ شبیه سازی مدار

ابتدا با استفاده از نرم افزار $proteus$ مدار موردنظر را طراحی و آزمایش کردیم، مراحل طراحی مدار به شرح زیر است:

۱.۲ مرحله اول: طراحی بلاک FA

با استفاده از دو گیت xor، دو گیت and و دو گیت or یک جمع کننده یک بیتی می سازیم، جدول درستی مدار داخل این بلاک به صورت زیر می شود:



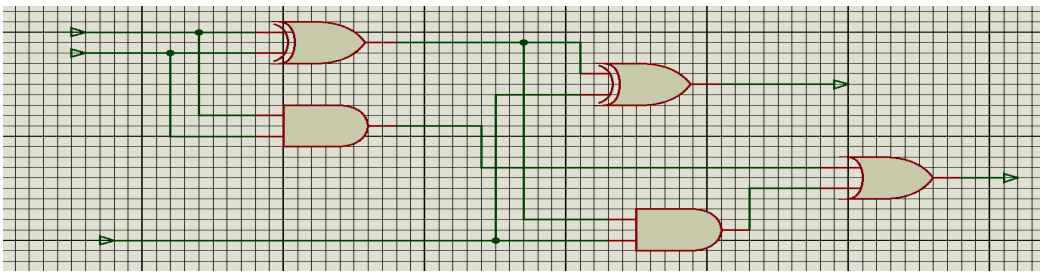
C_{in}	A	B	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

شکل ۲: جدول درستی $full\ adder$

که با توجه به جدول درستی بالا رابطه منطقی زیر را می توان در نظر گرفت:

$$S = A \oplus B \oplus C_{in}, \quad C_{out} = AB + C_{in}(A \oplus B)$$

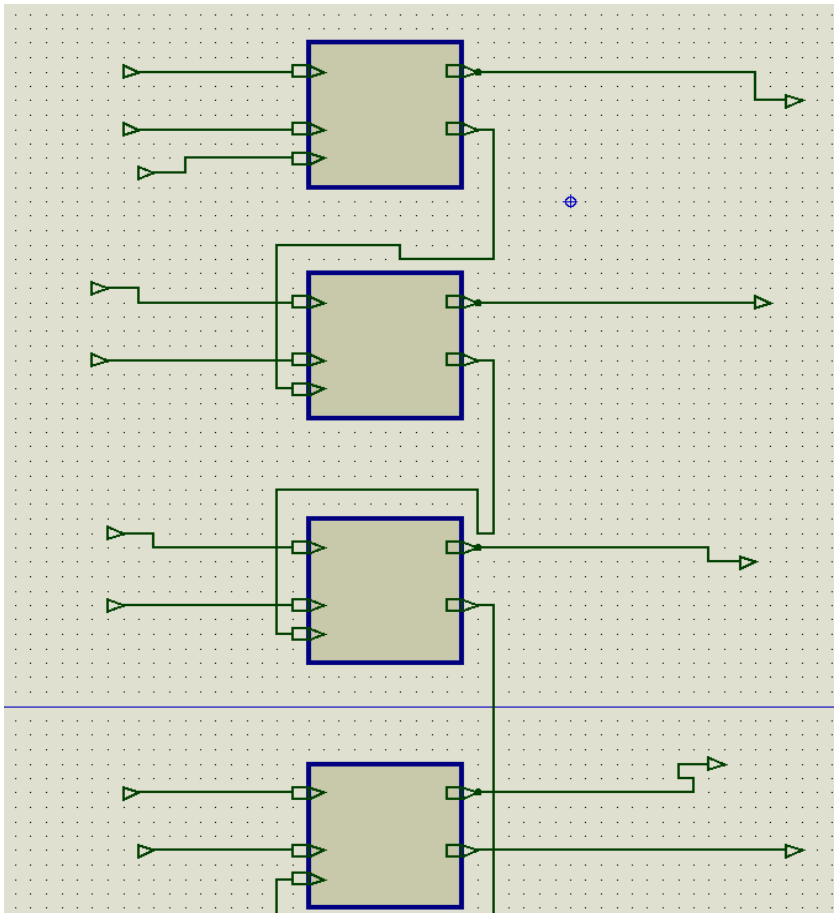
شکل داخلی این بلاک نیز و مدار داخل آن در نرم افزار $proteus$ به صورت می شود:



شکل ۳: مدار نهایی بلاک $full\ adder$

۲.۲ مرحله دوم: طراحی جمع کننده دهدهی یک رقمی

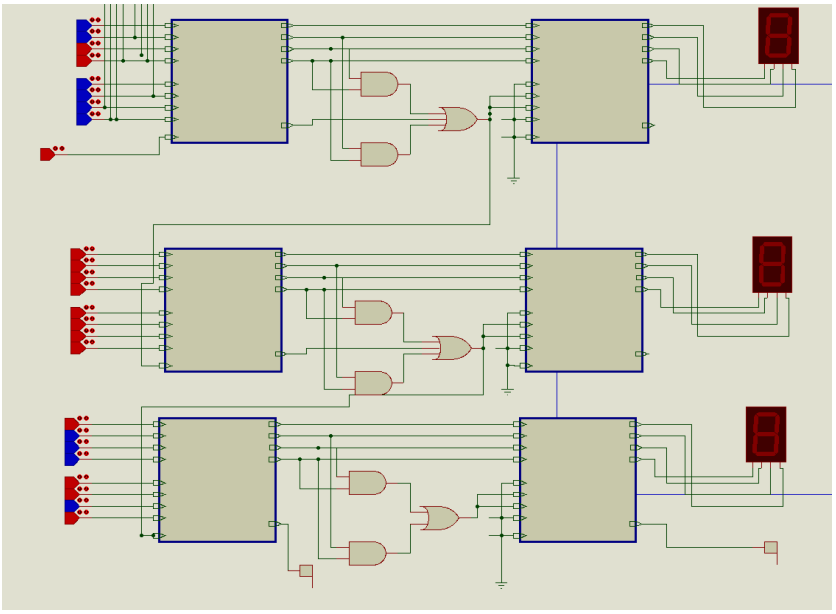
در این مرحله هر رقم را به صورت BCD در نظر میگریم، پس برای نمایش یک رقم دهدهی نیاز به ۴ بیت داریم، با استفاده از تمام جمع کننده هایی که در مرحله قبل ساختیم یک تمام جمع کننده دهدهی میسازیم، داخل این بلاک به صورت زیر است(هر بلاکی که در شکل آورده شدهه یک FA است)



شکل ۴: مدار نهایی بلاک *bcd adder*

۳.۲ مرحله سوم: طراحی یک تمام جمع کننده دهدهی سه رقمی

ابتدا دقت کنید که برای جمع دو عدد تک رقمی دهدهی از دو بلاک $1-digit\ BCD\ Adder$ استفاده می کنیم چرا که اگر جمع دو عدد از ۱۰ بیش تر شود و به اصطلاح $overflow$ کند باید حاصل با ۶ جمع شود تا در نهایت خروجی به درستی نمایش داده شود(درواقع عدد بزرگ تر از ۹ در یک رقم دسیمال جا نمی شود و باید ۱۰ واحد از حاصل کم کنیم که این معادل جمع کردن با ۶ است.)
برای پیاده سازی این قسمت از دو گیت or و یک گیت and استفاده کردیم به این صورت که اگر بیت دوم و سوم و یا اول و سوم خروجی $bcd-adder$ اول یک بود یعنی $overflow$ رخ داده است و حاصل خروجی بلاک اول را با ۶ جمع می کنیم و $carry$ جمع کننده دهدهی رقم بعدی را یک می کنیم و اگر این اتفاق نیافتاده بود خروجی بلاک اول را مستقیم نمایش می دهیم، همچنین برای ساده سازی نمایش خروجی از $7-segment-BCD$ استفاده کردیم، شکل نهایی مدار را در قسمت زیر مشاهده می کنید:

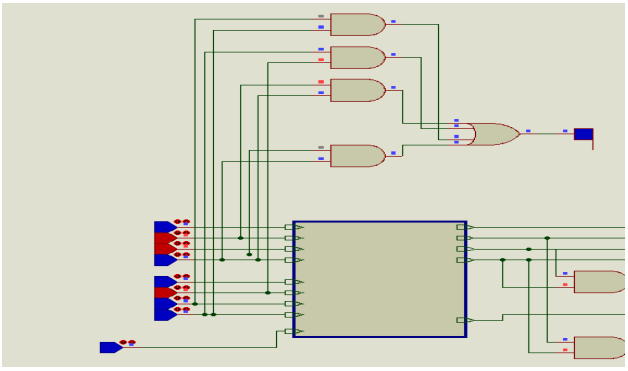


شکل ۵: مدار نهایی $3-digit\ bcd\ adder$

۴.۲ مرحله چهارم: ساخت چراغ هشدار دهنده

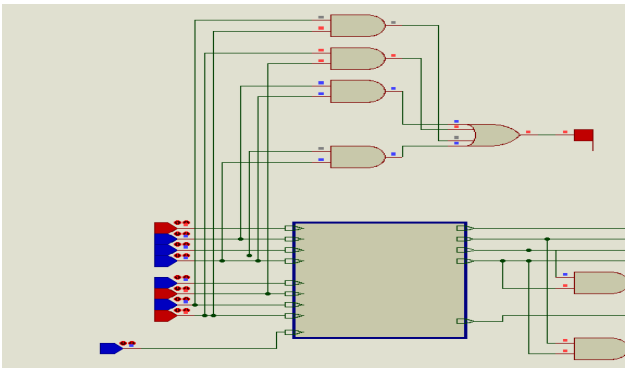
اگر هر رقم ورودی از ۹ بیش تر باشد قاعدتا به عنوان یک رقم دهدهی صحیح نیست، برای نشان دادن وضعیت صحیح بودن ورودی و یا خراب بودن آن از یک *logic probe* استفاده کردیم که اگر قرمز باشد به معنی خطا در ورودی است، به طوری کلی نحوه پیاده سازی آن نیز به این صورت است که برای هر رقم دهدهی بیت سوم و دوم و بیت سوم و اول آن را *and* می کنیم و سپس حاصل را *or* می کنیم اگر جواب نهایی یک بود یعنی رقم مورد نظر از ۹ بیش تر است، به طوری کلی نیز *or* نتایج تمام ورودی ها را خروجی نهایی در نظر می گیریم، در شکل های زیر دو حالت صحیح و خراب از ورودی ها آزمایش شده است و درستی عملکرد چراغ هشداردهنده بررسی شده است:

- یک رقم ۳ و دیگری ۲ باشد، در این حالت ارقام درست اند و خروجی نهایی ۰ و آبی است



شکل ۶: تست اول چراغ هشداردهنده

- یک رقم ۱ و دیگری ۱۰ باشد، در این حالت واضحا یک رقم نمی تواند ۱۰ باشد و ورودی نادرست است پس خروجی نهایی ۱ و قرمز است

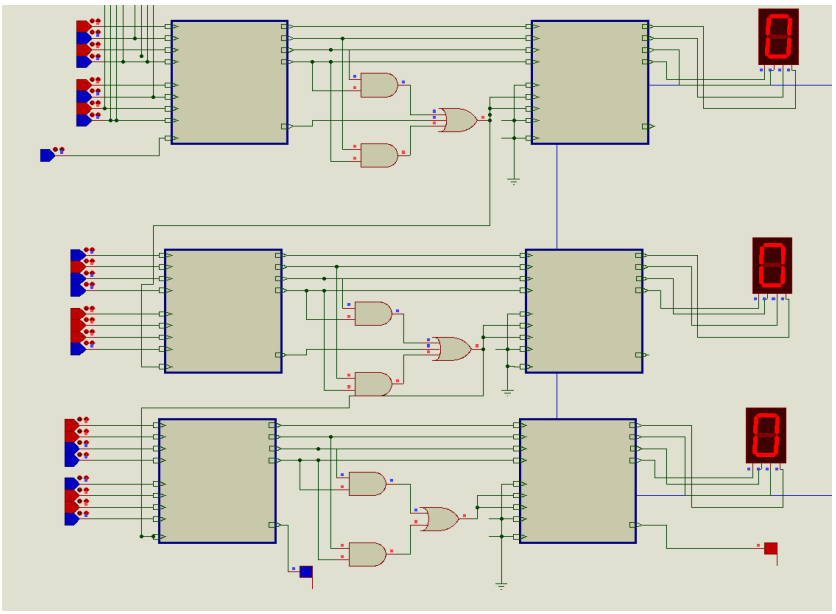


شکل ۷: تست دوم چراغ هشداردهنده

۵.۲ آزمایش مدار و بررسی درستی عملکرد آن

در این قسمت حالت های مختلف ورودی را به مدار دادیم و خروجی آن را بررسی کردیم، نتیجه سه تست را نیز در این قسمت آورده ایم:

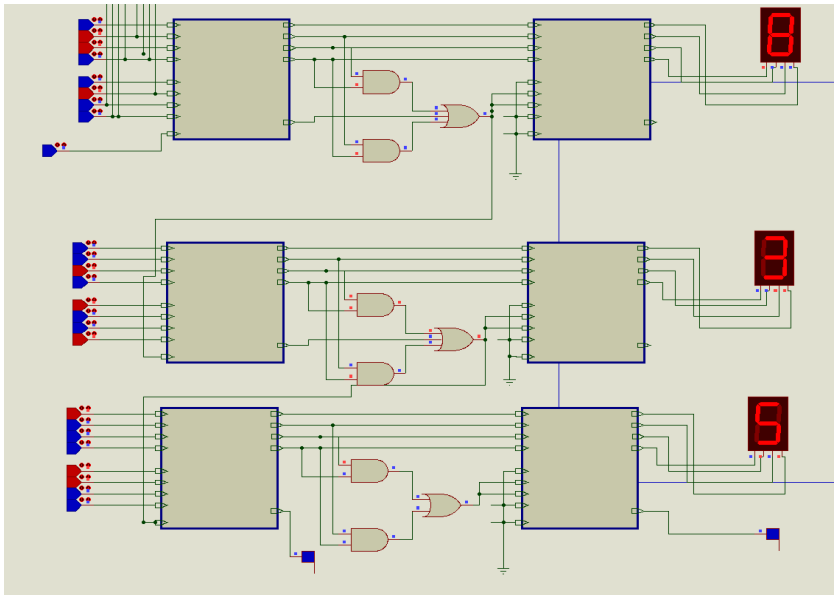
تست اول) ۱۰۰۰ = ۳۲۵ + ۶۷۵، که همانطور که می بینید بیت $carry$ تنها روشن است و خروجی نهایی مدار عدد ۱۰۰۰ را نشان می دهد:



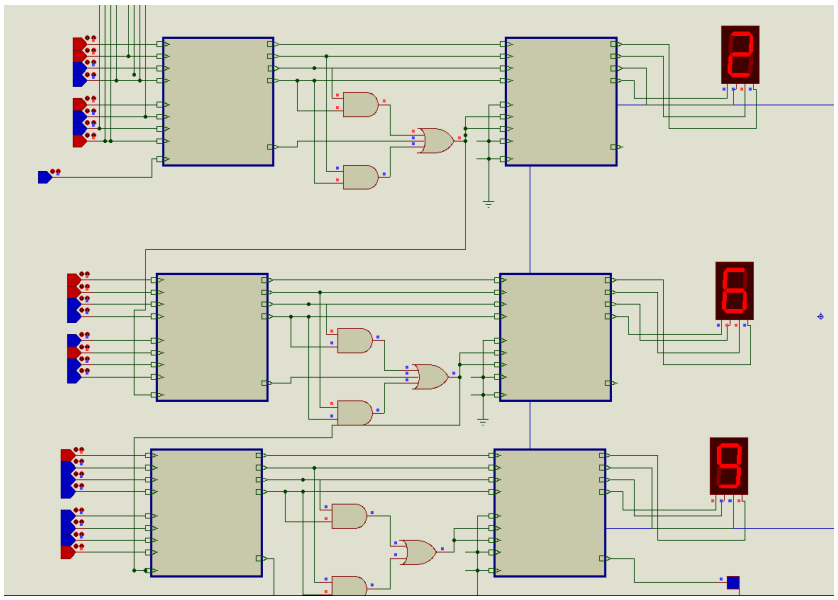
شکل ۸: تست اول مدار جمع کننده سه رقمی

تست دوم) ۵۳۸ = ۳۹۲ + ۱۴۶، که همانطور که می بینید خروجی مدار نیز ۵۳۸ را نشان می دهد:

تست سوم) ۹۶۲ = ۱۳۳ + ۸۲۹، که همانطور که می بینید خروجی مدار نیز ۹۶۲ را نشان می دهد:



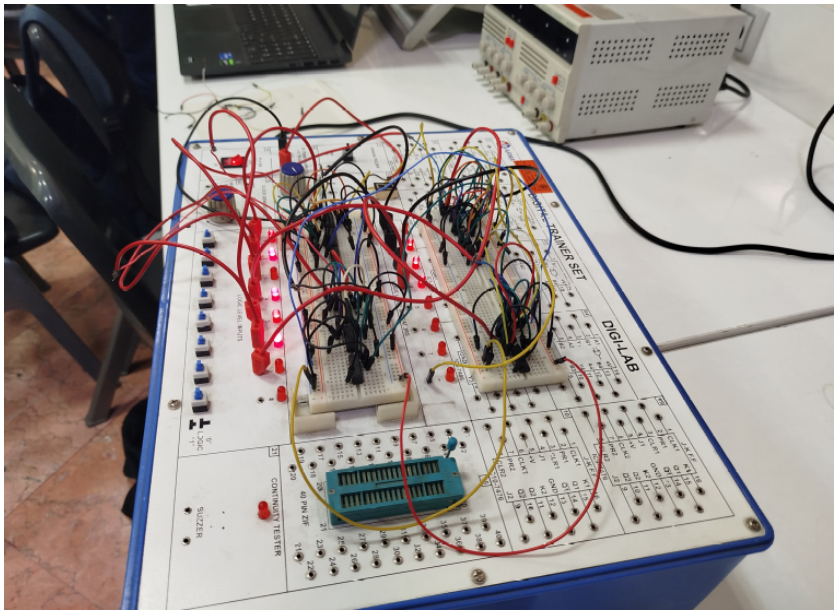
شکل ۹: تست دوم مدار جمع کننده سه رقمی



شکل ۱۰: تست سوم مدار جمع کننده سه رقمی

۳ پیاده سازی فیزیکی مدار در آزمایشگاه

ابتدا با استفاده از سه تراشه $7432, 7408, 7486$ یا همان xor, or, and دوتایی یک تمام جمع کننده یک بیتی روی بردبورد درست کردیم، سپس به طریق مشابه ۴ تا FA درست کردیم به طوری که روی هر بردبورد دو $full\ adder$ قرار دادیم و $carry$ هر کدام را به ترتیب به C_{in} تمام جمع کننده بعدی دادیم و به این صورت یه جمع کننده ۴ بیتی درست کردیم، شکل نهایی این جمع کننده به صورت زیر است:

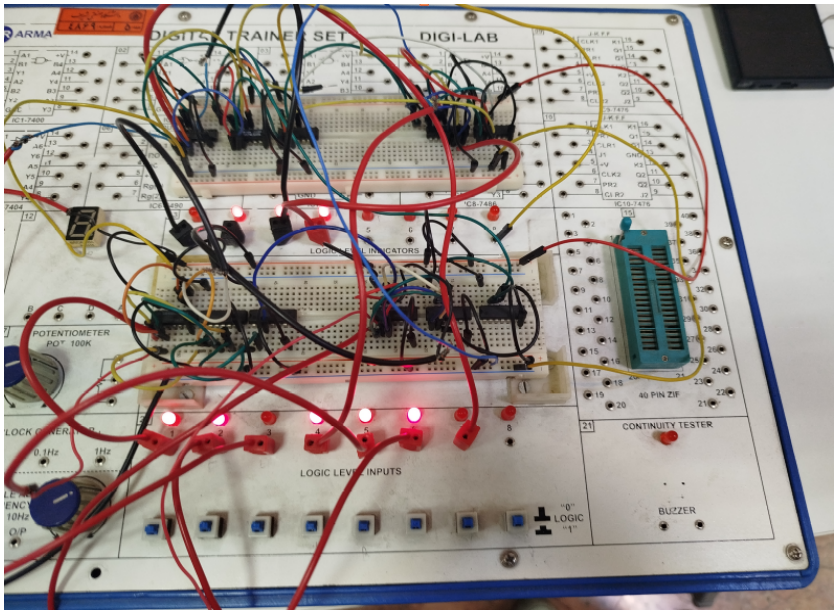


شکل ۱۱: مدار جمع کننده ۴ بیتی به روی بردبورد

و در شکل زیر نیز نتیجه یک تست روی این مدار را مشاهده می کنید که اگر دو عدد ورودی را A و B و خروجی نهایی را S بنامیم و همچنین $carry$ ورودی و خورجی C_{in} و C_{out} باشد داریم:

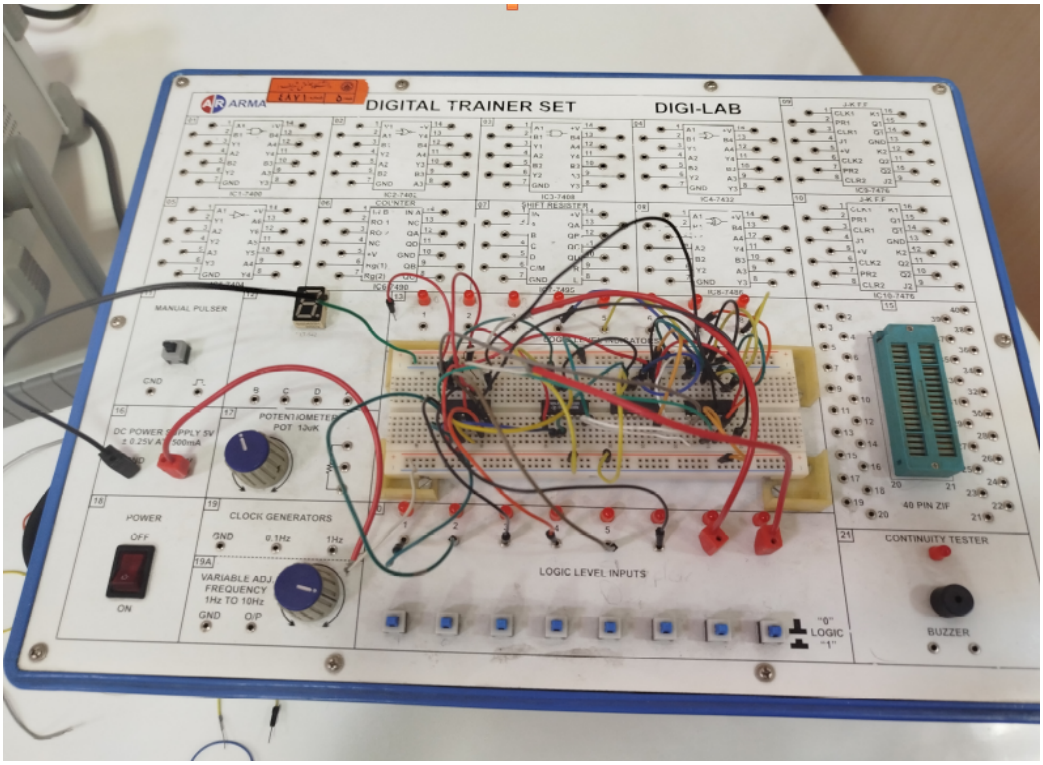
$$A = 0101,\quad B = 1001,\quad C_{in} = 1,\quad S = A + B + C_{in} = 1110,\quad C_{out} = 0$$

و همانطور که در شکل زیر مشخص است نتیجه منطقی بالا روی مدار ساخته شده نیز به وجود آمد و مدار جمع کننده ۴ بیتی ساخته شده درست کار می کند.



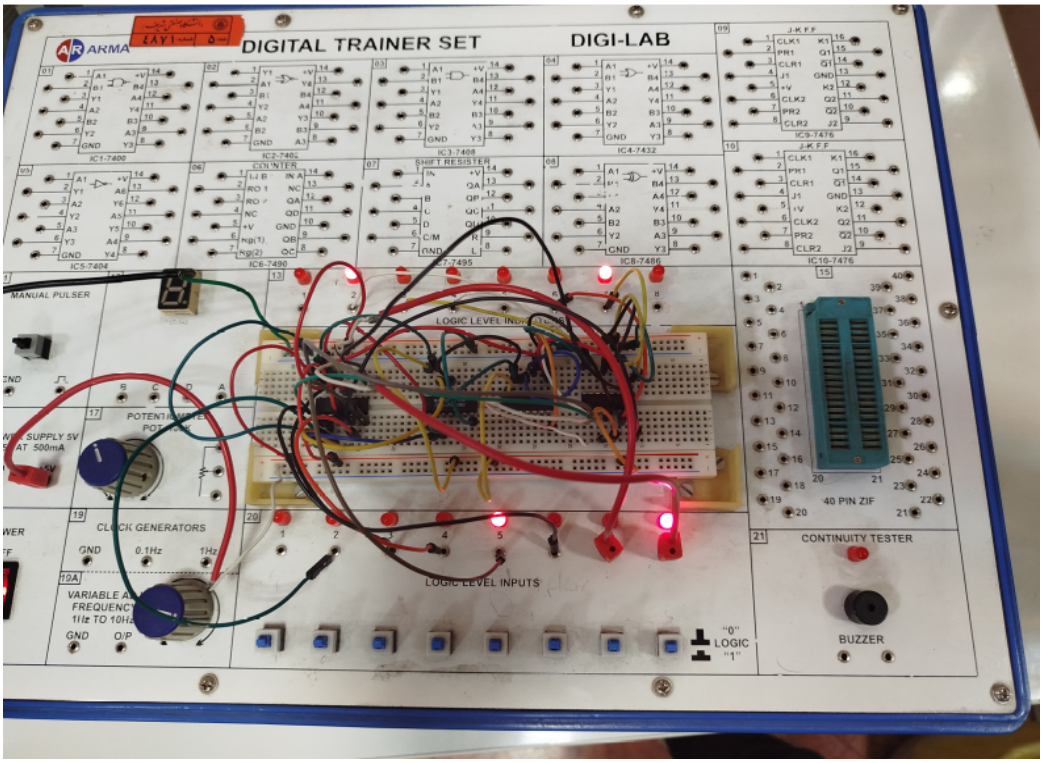
شکل ۱۲: تست مدار جمع کننده ۴ بیتی به روی بردبورد

در قسمت دوم نیز از دو تراشه $4 - bit\ adder$ استفاده کردیم و مطابق قواعدی که بالا تر در شبیه سازی گفته شد، یک جمع کننده دهدهی با استفاده از دو تراشه 7483 و تراشه های or و and دو ورودی ساختیم، شکل نهایی مدار به صورت زیر است:



شکل ۱۳: مدار جمع کننده دهدهی تک رقمی روی بردبورد

و در شکل زیر نیز نتیجه یک تست روی این مدار را مشاهده می کنید ک دو ورودی یکی ۹ و دیگری ۲ داده شده است خروجی ۱۱ شده است(در قسمت ورودی به ترتیب از راست به چپ بیت کم ارزش عدد دوم و سپس بیت کم ارزش عدد اول آمده است، در قسمت خروجی نیز بیت های سمت راست تر کم ارزش تراند):



شکل ۱۴: تست مدار جمع کننده دهدهی تک رقمی روی بردبورد

۴ چالش ها

عمده چالش هایی که در آزمایشگاه به وجود آمد را می توان به دو دسته کلی تقسیم کرد:
۱) بسیاری از قطعات و تراشه ها سالم نبودند، به طور مثال هنگام استفاده از بردبورد ابتدا که آن را با مولتی متر تست کردیم سالم بود اما هنگام استفاده در بعضی از خانه ها مشکل داشت، همین طور بعضی از پایه های تراشه ها و بعضا حتی خود تراشه ها سالم نبودند و باعث نتیجه خطا در خروجی و تست ها می شدند.
۲) اشتباهات فردی، به طور کلی حین آزمایش پیش میامد که به دلیل رنگ سیم ها دچار اشتباه می شدیم و یا یک خروجی را به پایه مناسب وصل نکرده بودیم.

۵ نتیجه و بحث

در این آزمایش با استفاده از گیت های ابتدایی $xor,\ or,\ and$ یک $full\ adder$ سپس همانطور که گفته شد یک $1-digit\ bcd\ adder$ ساختیم و در $proteues$ نیز با استفاده از بلاک های جمع کننده دهدهی یک بیتی یک جمع کننده دهدهی سه رقمی ساختیم و با تست حالات مختلف در هر قسمت از درستی آن مطمئن شدیم، به طور کلی دیدیم که استفاده از یک سلسله مراتب و انجام مرحله به مرحله باعث کاهش خطا و ساده تر شدن پیاده سازی می شود.