

بسمه تعالی



گزارش کار سوم آزمایشگاه معماری

جمع / تفریق کننده ممیز شناور

استاد:

دکتر حمید سربازی آزاد

نویسندگان:

امیررضا آذری ۹۹۱۰۱۰۸۷

غزل طحان ۹۹۱۰۶۳۷۴

بزرگمهر ضیا ۹۹۱۰۰۴۲۲

دانشگاه صنعتی شریف

تابستان ۱۴۰۲

هدف	۳
بخش اول _ ساخت مسیر داده	۳
بخش دوم _ ساخت واحد کنترل	۱۳
بخش سوم _ تکمیل مدار و تست آن	۱۷
نتیجه گیری:	۲۵
منابع و مراجع:	۲۶

هدف

هدف از این آزمایش آشنایی با نحوه کار یک جمع/تفریق کننده ممیز شناور می باشد. در این آزمایش دو عدد ۱۲ بیتی ممیز شناور در فرمت IEEE۷۵۴ به مدار می دهیم و پس از ۱ کردن سیگنال شروع و مشخص کردن عملیات (جمع یا تفریق) پس از گذشت چند کلاک، نتیجه مورد انتظار که حاصل انجام عملیات روی آن هاست، در خروجی مشاهده می شود. همچنین مدار خروجی های زیر را دارد:

Overflow: وقوع یا عدم وقوع سرریز را مشخص می کند.

Infinity: اگر یکی از ورودی ها برابر با بی نهایت شود، ۱ شده و خروجی بی نهایت نشان داده می شود.

NaN: اگر یکی از ورودی ها معتبر نباشد، ۱ شده و خروجی دیگر مهم نخواهد بود.

نکته اینکه این مدار فقط تحویل پروتئوس دارد و کاردر کلاس برای آن تعریف نشده است.

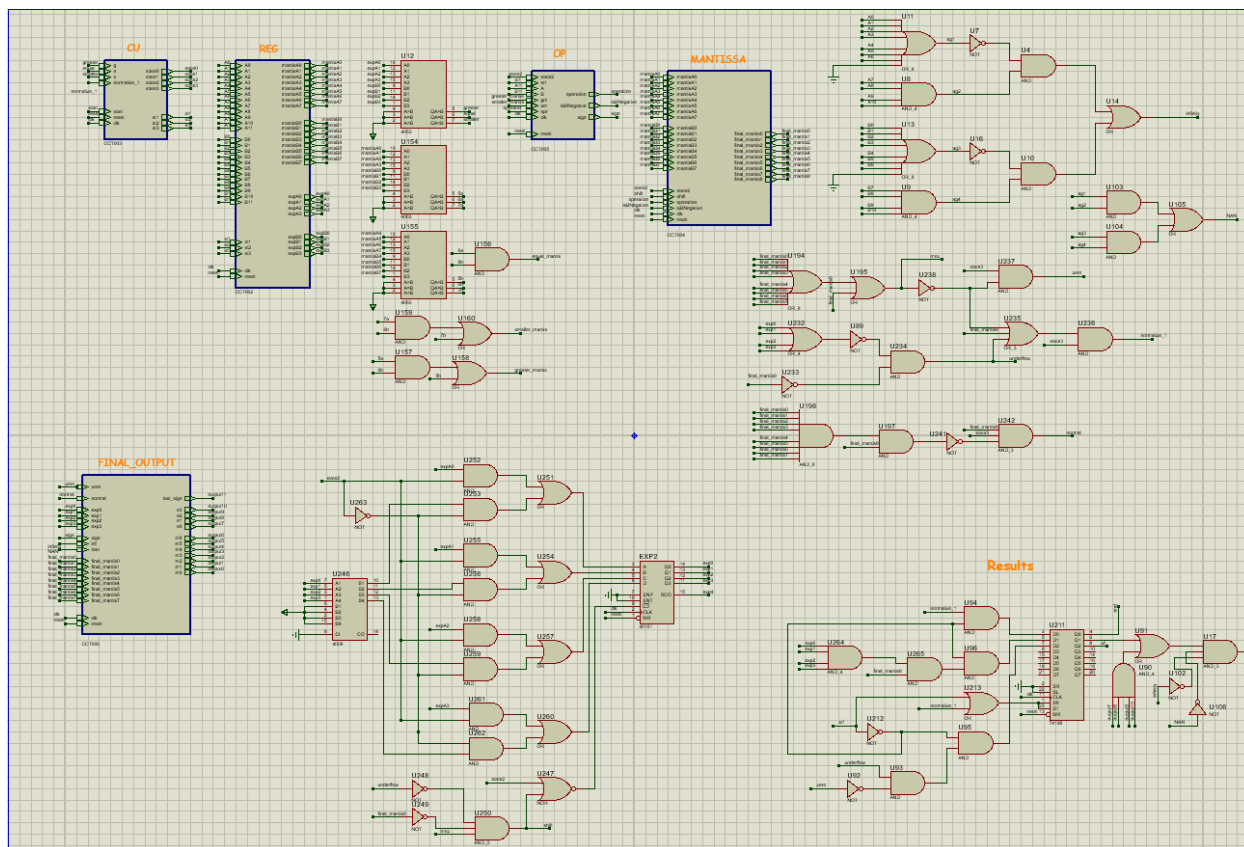
بخش اول _ ساخت مسیر داده

تئوری آزمایش:

در این بخش سعی داریم مسیر داده مدار را طراحی کرده و آن را به واحد کنترل متصل نماییم. الگوریتم مسیر داده از مراحل زیر تشکیل شده است:

۱. ورودی ها در رجیستر لود می شوند تا با تغییر آنها مشکلی در مدار پیش نیاید.
۲. برای جمع یا تفریق اعداد ممیز شناور باید توان آنها برابر باشد. در نتیجه عدد با توان کمتر، قسمت مانتیس آن را آنقدر شیفت داده و به قسمت توان آن اضافه می کنیم تا با توان عدد بزرگتر یکی شود.
۳. با توجه به علامت اعداد و عملیات درخواستی در ورودی، عملیات روی مانتیس ها انجام شده و جواب نهایی آماده شده اما در خروجی نشان داده نمی شود.
۴. در صورتی که جواب نهایی نرمال نباشد (قسمت مانتیس آن به فرمت ۱.X نباشد)، نرمال شده و خروجی overflow نیز مشخص می شود.
۵. جواب نهایی در خروجی نشان داده شده و سیگنال end ۱ می شود.

گزارش کار بخش اول :

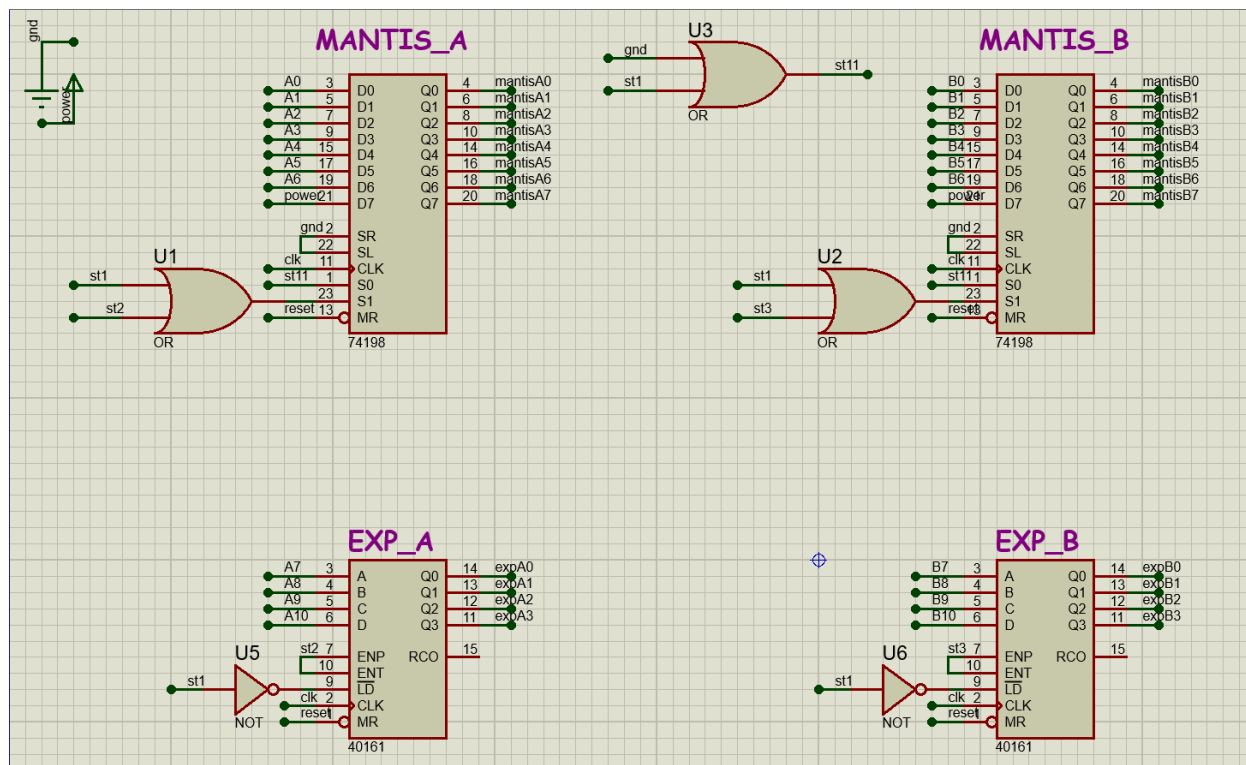


شکل ۱. ساختار مسیر داده مدار

در شکل ۱ ساختار مسیر داده مدار را مشاهده می کنید. در طراحی این ساختار از تعدادی کامپوننت استفاده شده که در ادامه هر یک را توضیح می دهیم:

۱-۱: رجیسترها (REG)

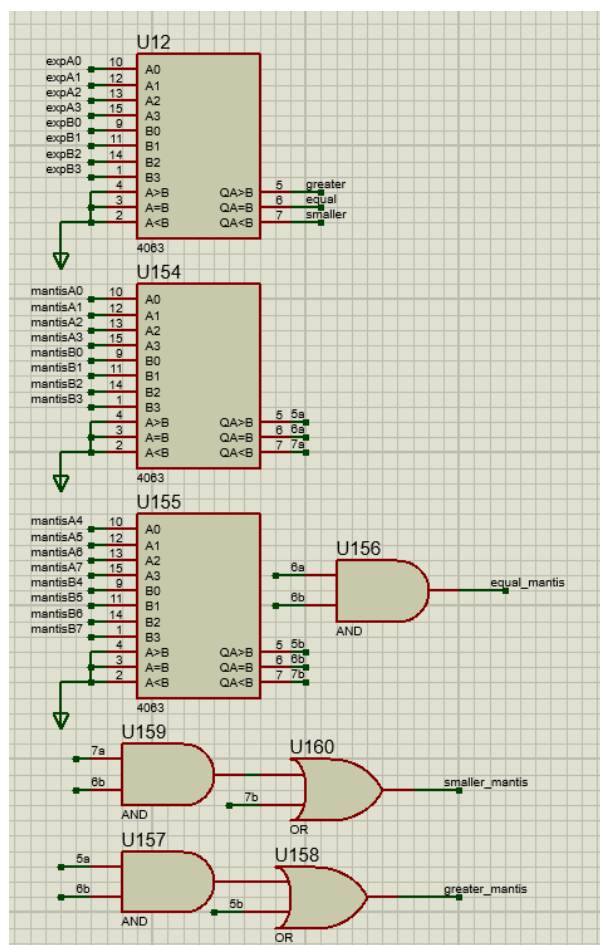
این مدار از دو عدد شیفت رجیستر ۷۴۱۹۸ برای نگه داری و شیفت مانتیس ها و دو عدد شمارنده ۴۰۱۶۱ برای نگه داری و تغییر توان ها تشکیل شده است. سیگنال st_1 ورودی لود است که از واحد کنترل می آید و اگر ۱ باشد، مانتیس ها و توان اعداد در رجیسترها لود می شوند. به همین دلیل این ورودی به ورودی های S_0 ۷۴۱۹۸ و LD' ۴۰۱۶۱ متصل شده است. سیگنال های S_2 و S_3 برای کنترل شیفت مانتیس ها و افزایش توان استفاده می شوند. به این صورت که هرکدام که ۱ شوند، مانتیس مربوط به آن به سمت راست شیفت داده شده و توان مربوط به آن افزایش می یابد. توجه کنید که سیگنال st_1 صرفا همان سیگنال st_1 است. همچنین سیگنال ریست مدار به ورودی ریست گیت ها متصل شده است.



شکل ۲. ساختار مدار رجیسترها

۲-۱: مدار مقایسه کننده

این مدار از سه عدد مقایسه کننده ۴ بیتی ۴۰۶۳ برای مقایسه مانتیس ها و توان ها تشکیل شده است. یکی از این مقایسه کننده ها برای مقایسه توان ها و دوتای بعدی برای مقایسه ۴ بیت کم ارزش و پر ارزش مانتیس ها استفاده می شود. سه خروجی equal_mantis، greater_mantis و smaller_mantis وضعیت مقایسه بین توان های A و B هستند. خروجی equal_mantis زمانی ۱ است که هر ۸ بیت دو مانتیس برابر باشد. خروجی smaller_mantis که نشانگر $\text{mantissa } A < \text{mantissa } B$ است، زمانی ۱ است که یا ۴ بیت پر ارزش مانتیس A کمتر از B باشد، یا ۴ بیت پر ارزش برابر بوده و ۴ بیت کم ارزش A از B کمتر باشد. همین حالات به طور مشابه برای خروجی greater_mantis که نشانگر $\text{mantissa } A > \text{mantissa } B$ است صدق می کنند.

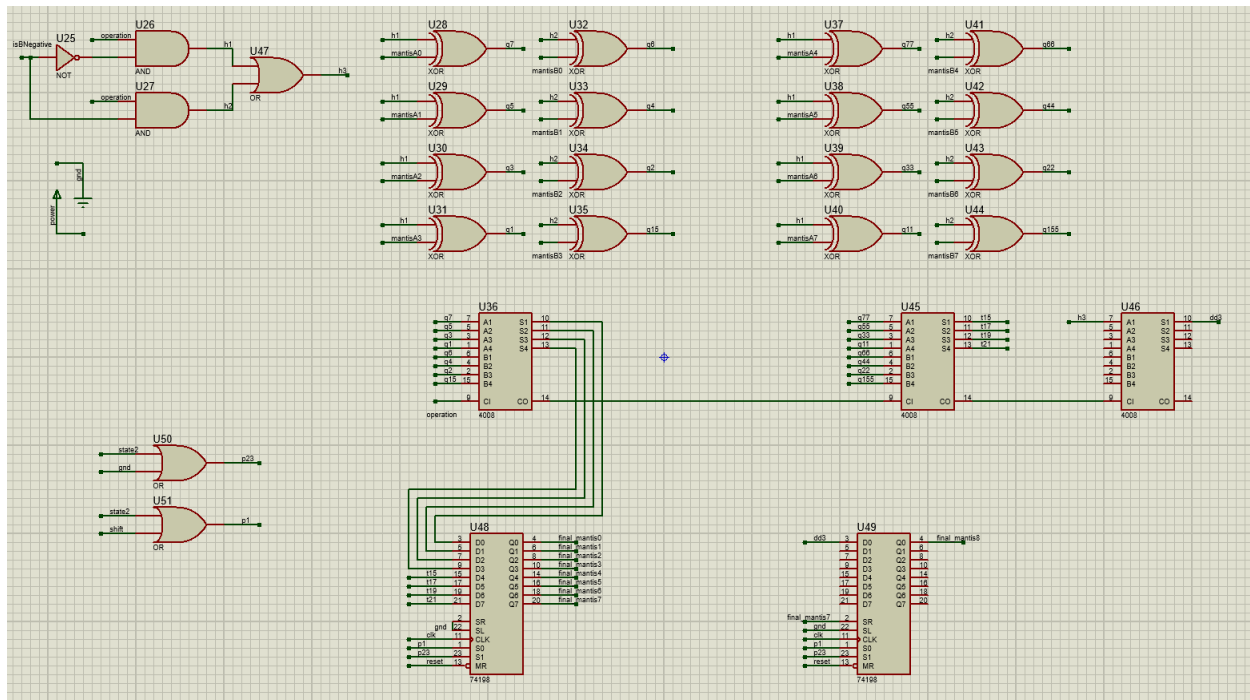


شکل ۳. ساختار مدار مقایسه کننده

۱-۳: مدار محاسبه مانتیس (MANTISSA)

این مدار وظیفه انجام عملیات مشخص شده در مدار OP را روی مانتیس‌ها دارد. ورودی‌های آن مانتیس‌های A, B, shift, isBNegative, operation, state^۲ هستند که دو مورد آخر از مدار OP می‌آیند. سیگنال کنترلی h^۱ زمانی ۱ است که operation = ۰ و isBNegative = ۰ باشد که اگر این اتفاق بیفتد، مانتیس A باید مکمل شود. سیگنال h^۲ نیز زمانی ۱ است که operation = ۱ و isBNegative = ۱ باشد که اگر این اتفاق بیفتد، مانتیس B باید مکمل شود. پس از این اتفاق، مانتیس‌ها به ورودی‌های جمع‌کننده‌های ۴ بیتی ۴۰۰۸ متصل می‌شوند. operation نیز به ورودی بیت نقلی جمع‌کننده بیت‌های کم ارزش داده می‌شود. خروجی جمع‌کننده نیز به ورودی شیفت رجیستر داده می‌شود تا در صورتی که در استیت state^۲ هستیم لود شوند و در صورتی که سیگنال shift ۱ شود (در هنگام نرمال کردن) به چپ شیفت داده شوند. سیگنال‌های p^۱ و p^{۲۳} برای همین هستند.

دیگر وظیفه این مدار، محاسبه بیت پرارزش مانتیس خروجی است که موقع محاسبه جمع مانتیس‌ها، این مقدار برابر جمع خروجی نقلی جمع‌کننده و $h^3 = h^1 + h^2$ است (dd^۳) و هنگام نرمال سازی نیز باید یک بیت شیفت داده شود.



شکل ۴. ساختار مدار محاسبه مانتیس

۴-۱: مدار Results

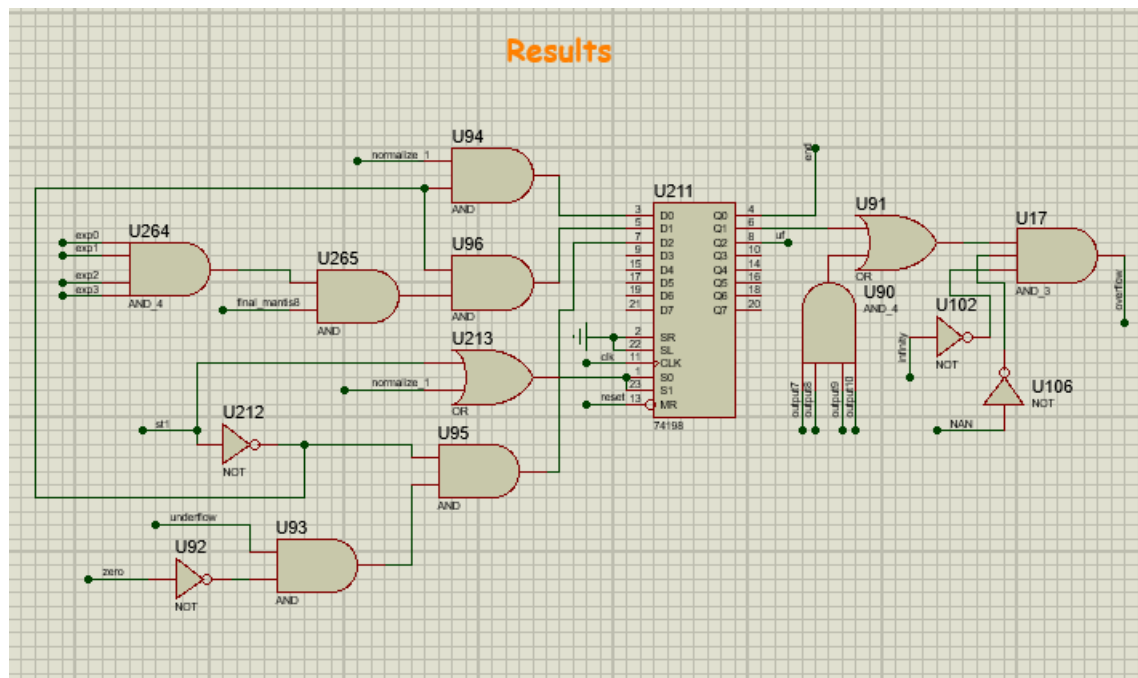
این مدار خروجی‌های overflow, underflow و end را تولید می‌کند. خروجی end در صورتی ۱ می‌شود که حاصل نرمال شده باشد و $st = 0$ باشد.

خروجی overflow زمانی ۱ است که Infinity و NaN برابر ۰ باشند و یکی از اتفاقات زیر بیفتد:

(۱) توان برابر ۱۱۱۱ و بیت پرارزش مانتیس برابر ۱ باشد و $st = 0$ باشد.

(۲) توان نهایی برابر ۱۱۱۱ باشد که این مورد نیازی به لود در رجیستر ندارد.

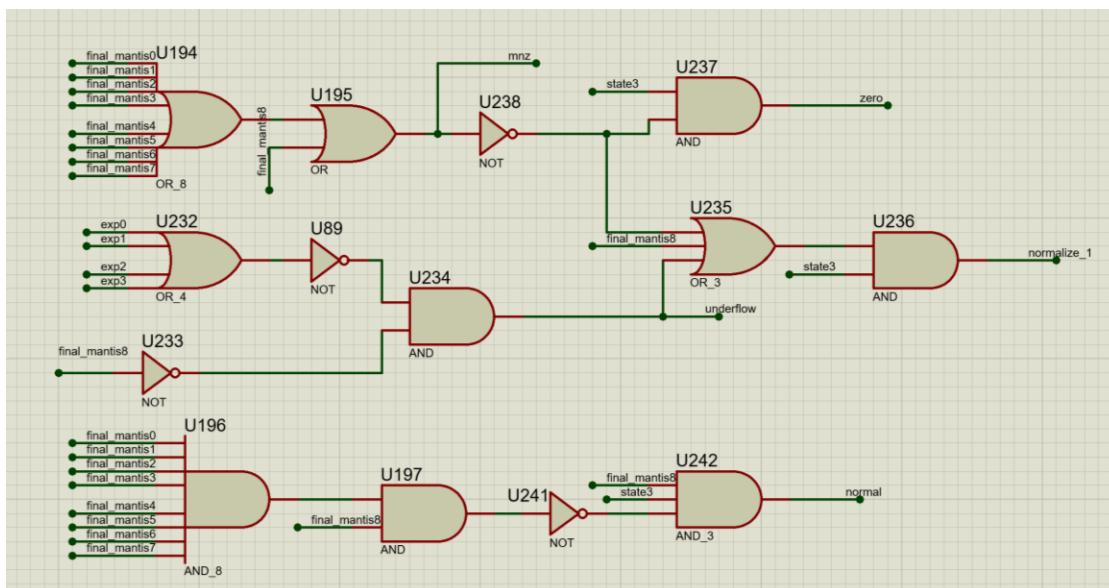
خروجی uf نیز زمانی ۱ است که $st = 0$ ، $zero = 0$ (حاصل ۰ نشده باشد) و $underflow = 1$ باشد. همچنین لود رجیستر زمانی ۱ است که $st = 1$ باشد یا خروجی نرمال شده باشد.



شکل ۵. ساختار مدار Results

۵-۱: مدار OP

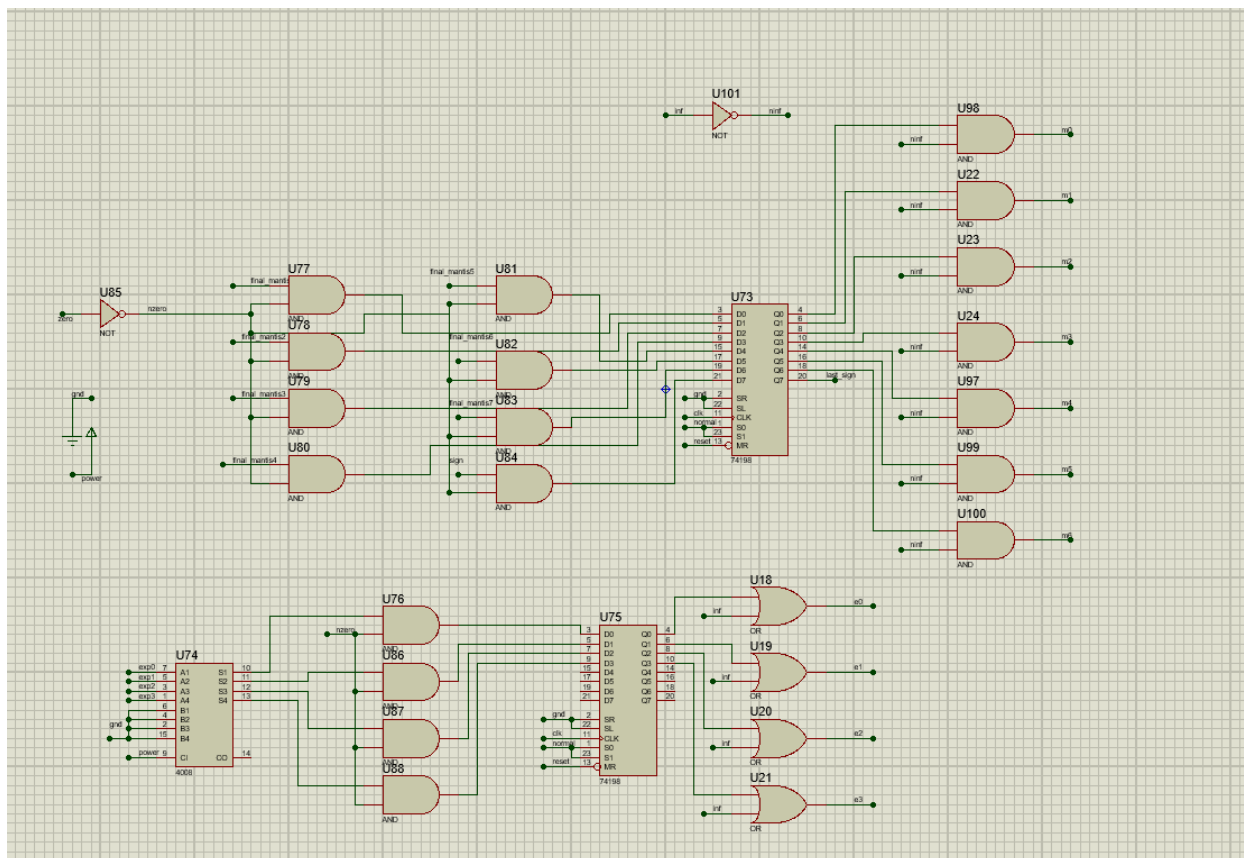
این مدار سه سیگنال `operation`, `isBNegative`, `sign` را محاسبه و تعیین می‌کند. `sign` علامت حاصل جمع/تفریق را مشخص می‌کند. `operation` مشخص می‌کند که چه عملیاتی بین مانتیس‌ها انجام شود و `isBNegative` مشخص می‌کند که نیاز هست B برای محاسبه ی حاصل، مکمل شود یا نه. منطق مدارهایی که برای هر کدام از سیگنال‌ها بسته شده، در شکل آمده و حالت‌های مختلف را پوشش داده‌ایم.



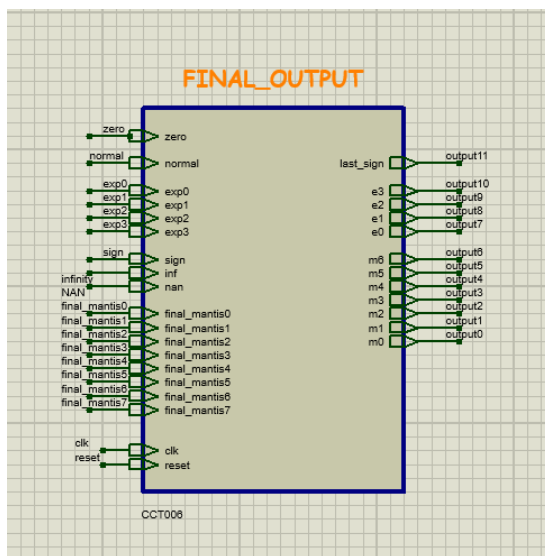
شکل ۷. مدار محاسبه کننده سیگنال های خروجی

۷-۱: مدار OUTPUT

این ماژول مانتیس، توان و علامت حاصل نهایی را خروجی می‌دهد. منطق این بخش از مدار به این صورت است که اگر حاصل صفر بود، طبق قرارداد، تمامی بیت‌های حاصل نهایی صفر شود و در غیر این صورت، حاصل نهایی همان ورودی ماژول خواهد بود و تنها به توان نهایی یکی اضافه می‌شود. همچنین در صورتی که سیگنال inf برابر ۱ باشد (یکی از ورودی‌ها بی‌نهایت باشد)، خروجی نیز برابر بی‌نهایت خواهد بود. مدار پیاده شده و ساختار کلی مدار این ماژول را مشاهده می‌کنید:



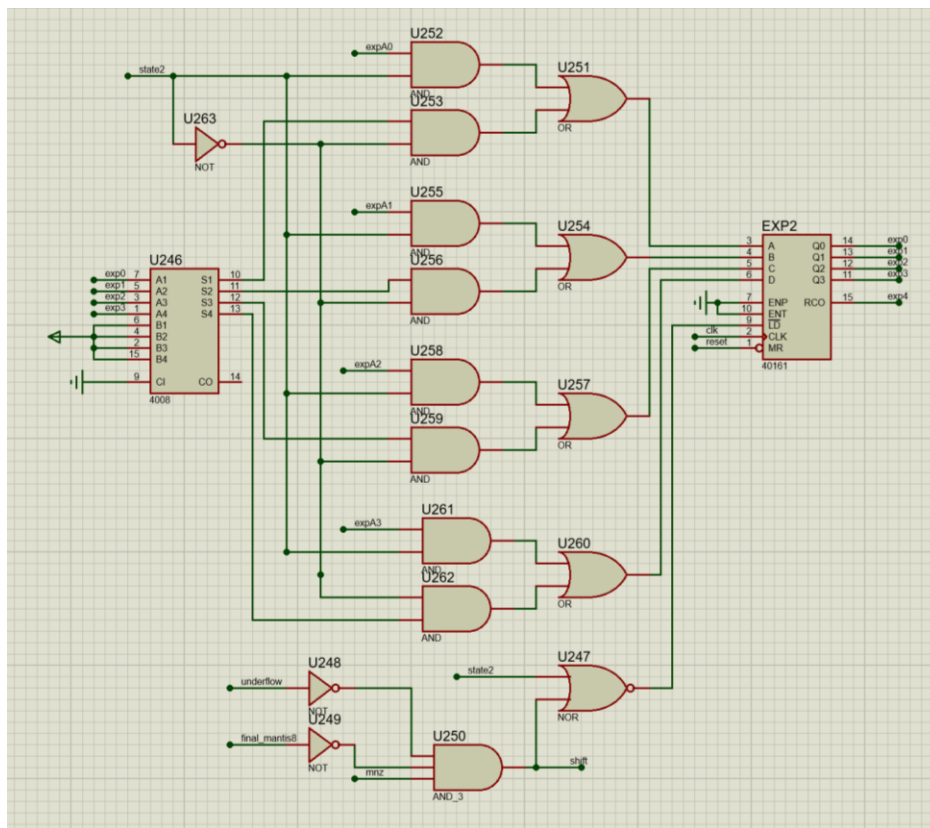
شکل ۸. ساختار مدار OUTPUT



شکل ۹. ساختار کلی مدار OUTPUT

۸-۱: محاسبه توان

از این ماژول در دو بخش از محاسباتمان بهره می‌گیریم. یکی زمانی که در استیت دوم هستیم و توان A را باید به عنوان توان نهایی لود کنیم و دیگر زمانی که در استیت نهایی و در حال نرمال‌سازی هستیم. با توجه به asm ای که رسم کردیم، در استیت سوم هر کجا لازم باشد از توان یکی کم می‌کنیم (که معادل با جمع کردن با ۱۱۱۱ است) و یا آن را شیفت می‌دهیم (که با شماره استیت، بیت نهم مانتیس و صفر نبودن مانتیس تعیین می‌شود). در شکل مدار آنرا مشاهده می‌کنیم:



شکل ۱۰. مدار محاسبه کننده توان

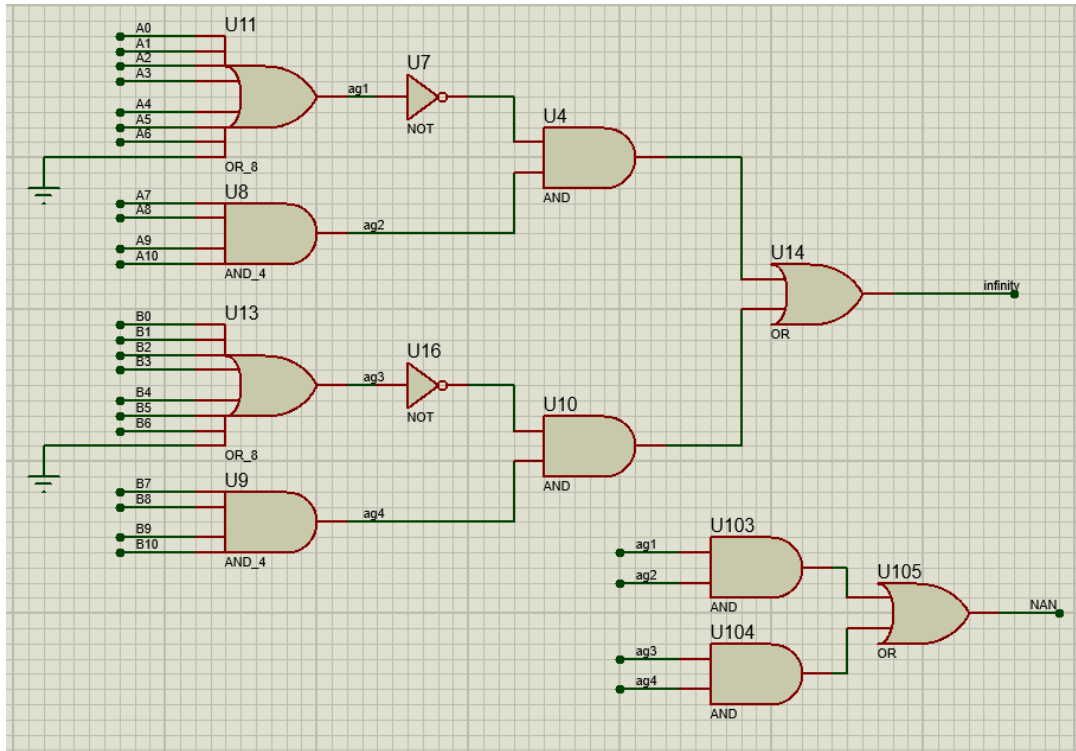
۹-۱: محاسبه infinity و NAN

از این ماژول برای محاسبه دو سیگنال خروجی infinity و NAN استفاده می‌کنیم. سیگنال infinity زمانی ۱ است که یکی از دو ورودی برابر یکی از اعداد زیر باشد:

$$0.1111 \dots / 1.1111 \dots$$

سیگنال NAN زمانی ۱ است که بخش توان برابر ۱۱۱۱ باشد اما بخش مانتیس برابر ۰ نباشد.

در شکل زیر تصویری از ماژول را مشاهده می‌کنید.

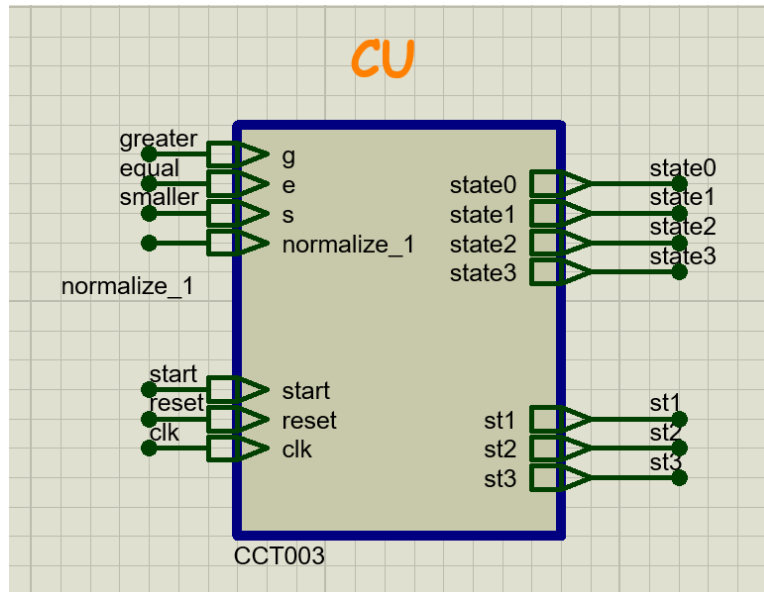


شکل ۱۱. ماژول محاسبه infinity و NaN

بخش دوم _ ساخت واحد کنترل

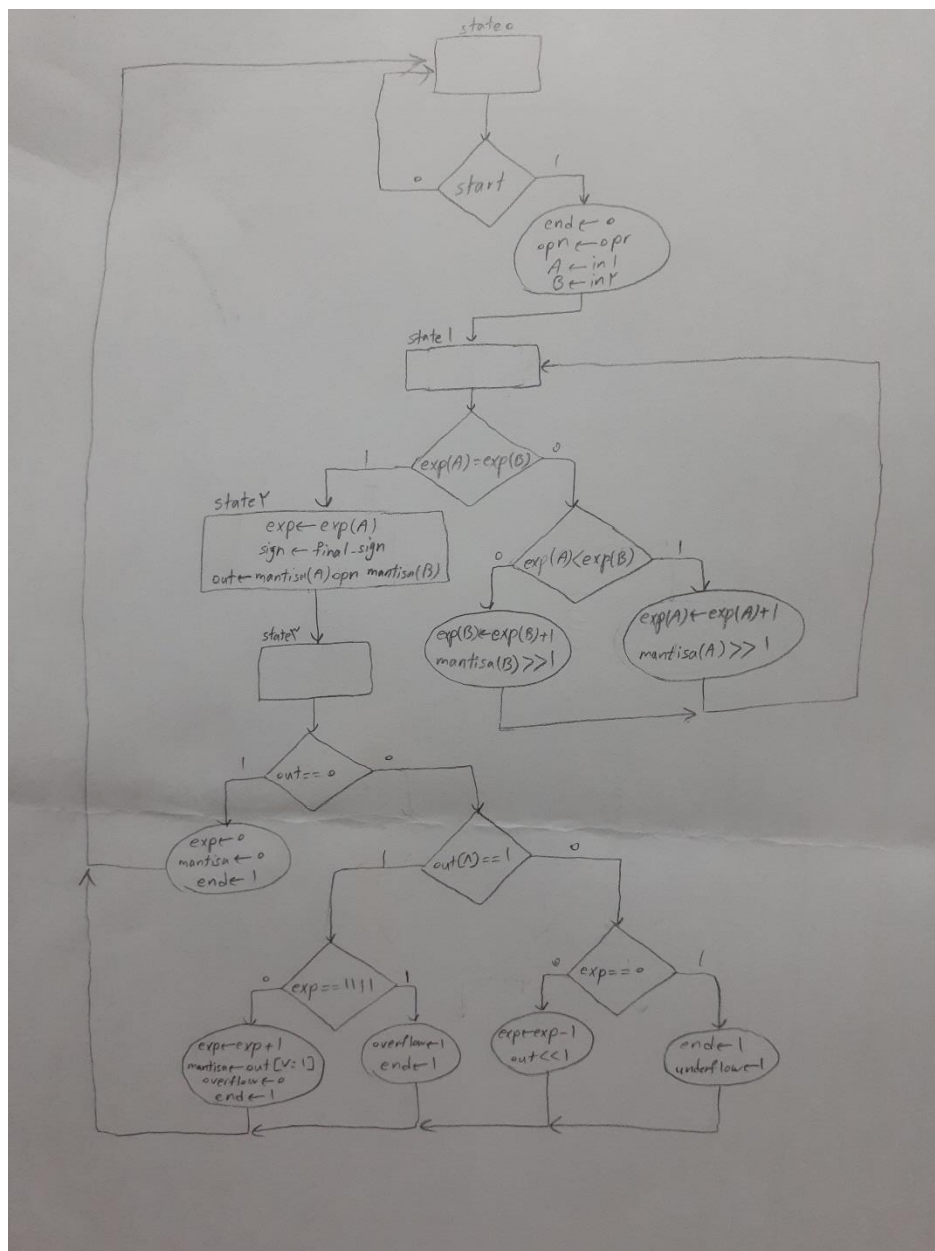
تئوری آزمایش:

در این بخش سعی داریم واحد کنترل مدار را طراحی کنیم. ورودی‌ها و خروجی‌های مدار مطابق شکل زیر هستند.



شکل ۱۲. ساختار کلی مدار واحد کنترل

این مدار مطابق asm chart زیر کار می کند و ۴ استیت دارد.



شکل ۱۳. طرح ASM Chart مدار

گزارش کار بخش دوم:

ساختار داخلی این واحد مطابق شکل زیر است.

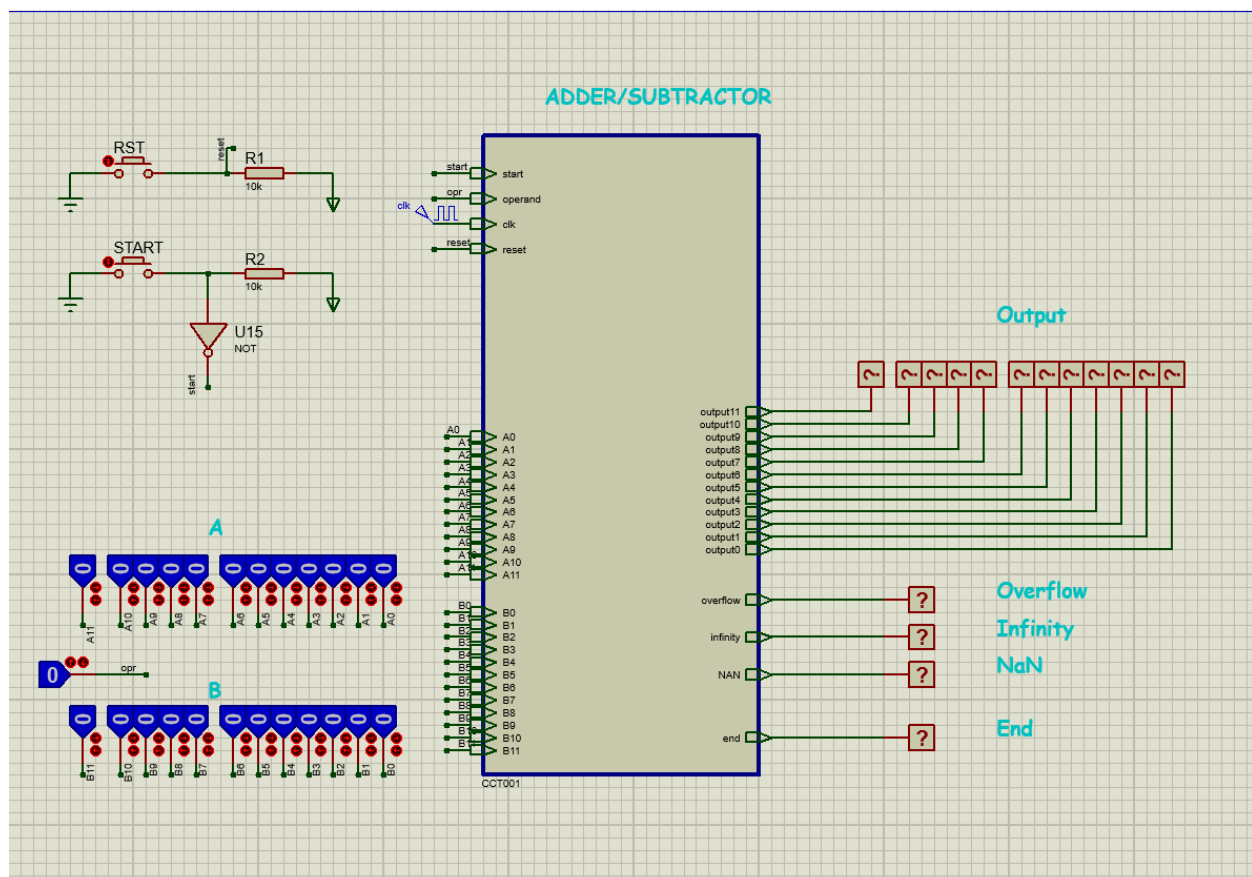
normalize_ \ : out == .

s , g : exp(A) < exp(B) == \ , .

بخش سوم _ تکمیل مدار و تست آن

تئوری آزمایش:

در بخش نهایی این آزمایش، به کمک بلوک های قبلی، به تکمیل طراحی مدار و تست کردن آن می پردازیم. مدار نهایی به شکل زیر می باشد.



شکل ۱۵. طرح کلی مدار نهایی

گزارش کار بخش سوم:

در این بخش، ورودی های A و B را همانطور که قابل مشاهده است، مشخص کرده ایم. ورودی های start و reset نیز به دکمه متصل شده اند که در صورت فشردن مدار شروع به کار می کند یا ریست می شود. ورودی کلاک نیز مشخص شده است.

ورودی opr نیز نشان‌دهنده عملیات جمع یا تفریق است. سپس تمامی بیت ها را به بلوک قبلی که در بخش پیشین به توضیح آن پرداخته‌ایم، می‌دهیم.

تست های زیر را انجام می‌دهیم.

به ترتیب تست ها را نمایش می‌دهیم. خروجی ها پس از اتمام کار مدار نمایش داده شده‌اند.

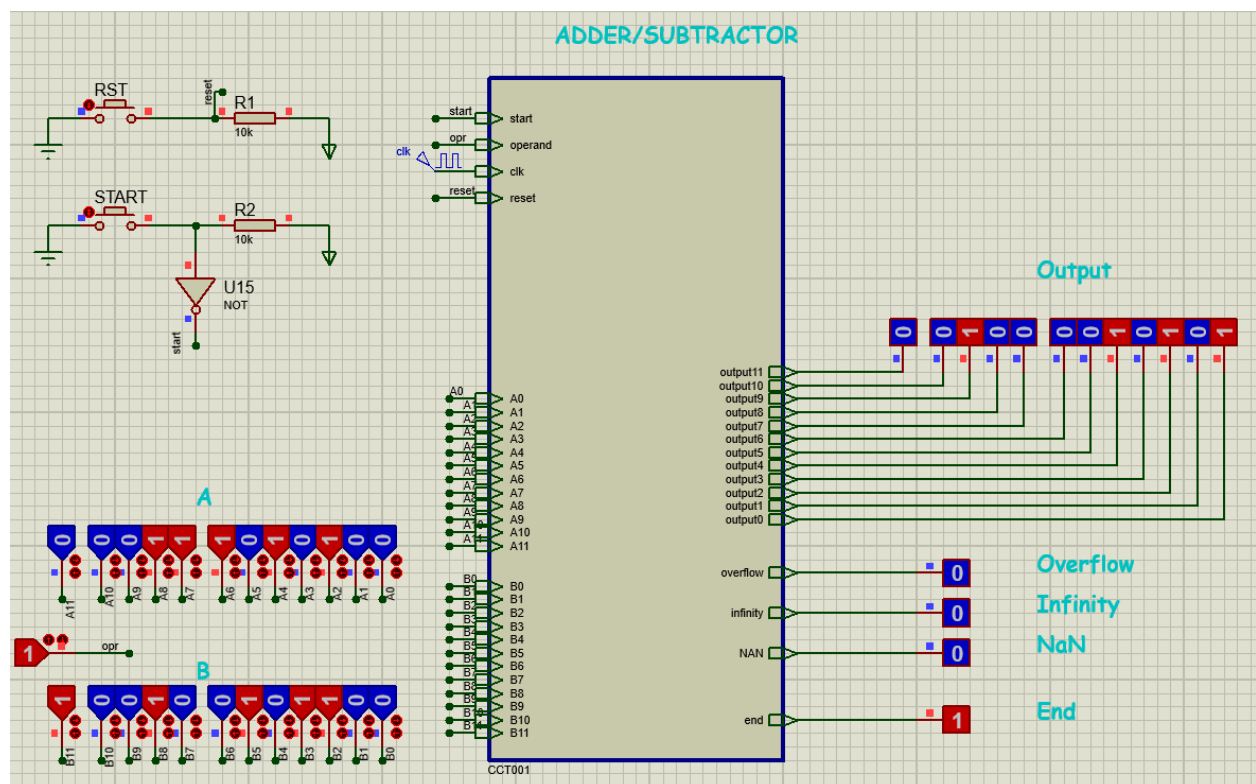
تست اول: (تفریق دو عدد مثبت و منفی)

$$A = 000111010100 = 0,103516$$

$$B = 100100101100 = -0,0419922$$

$$\text{opr} = 1 \text{ (Subtract)}$$

$$\text{Output} = 001000010101 = 0,1455082$$



شکل ۱۶. تست اول

همانطور که از مدار انتظار می‌رفت، خروجی مورد انتظار را مشخص کرده است.

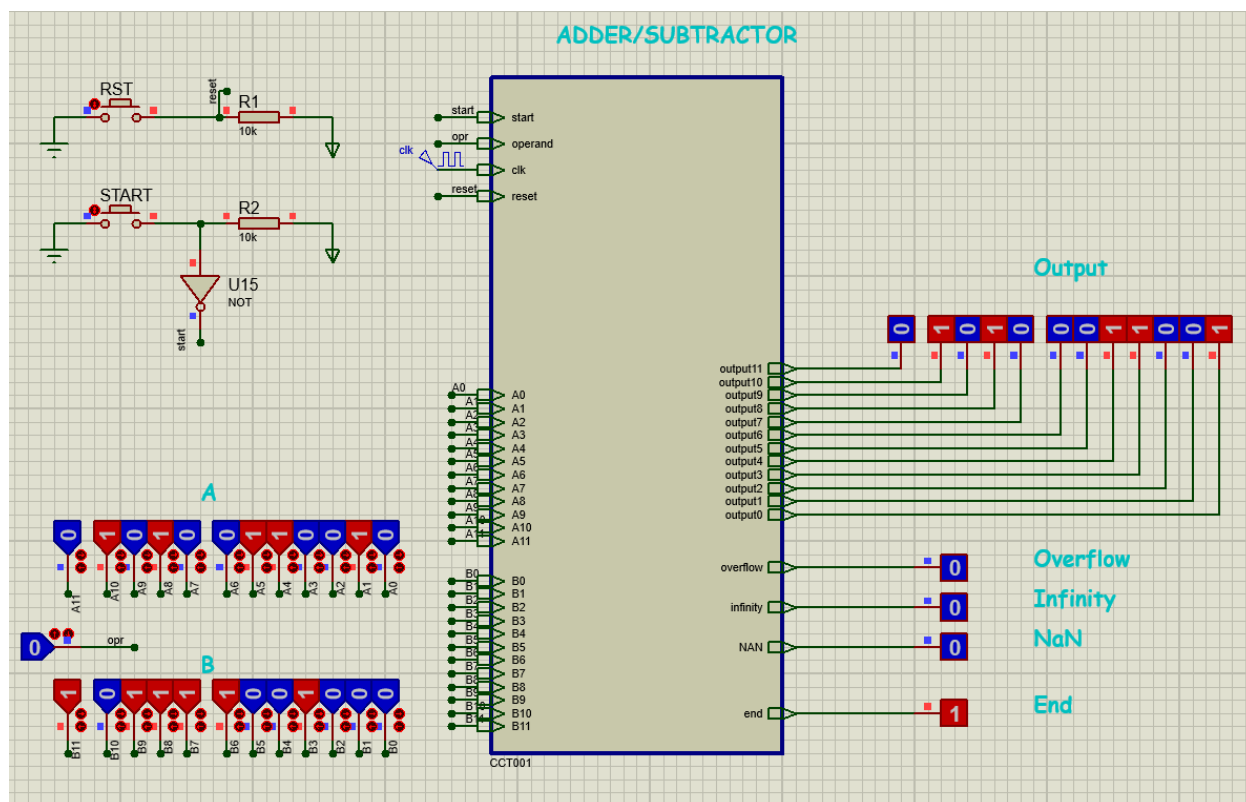
تست دوم: (جمع دو عدد مثبت و منفی)

$$A = 0101011010 = 11,125$$

$$B = 10111001000 = -1,0625$$

$$\text{opr} = 0 \text{ (Add)}$$

$$\text{Output} = 01010110101 = 9,0625$$



شکل ۱۷. تست دوم

همانطور که از مدار انتظار می‌رفت، خروجی مورد انتظار را مشخص کرده است.

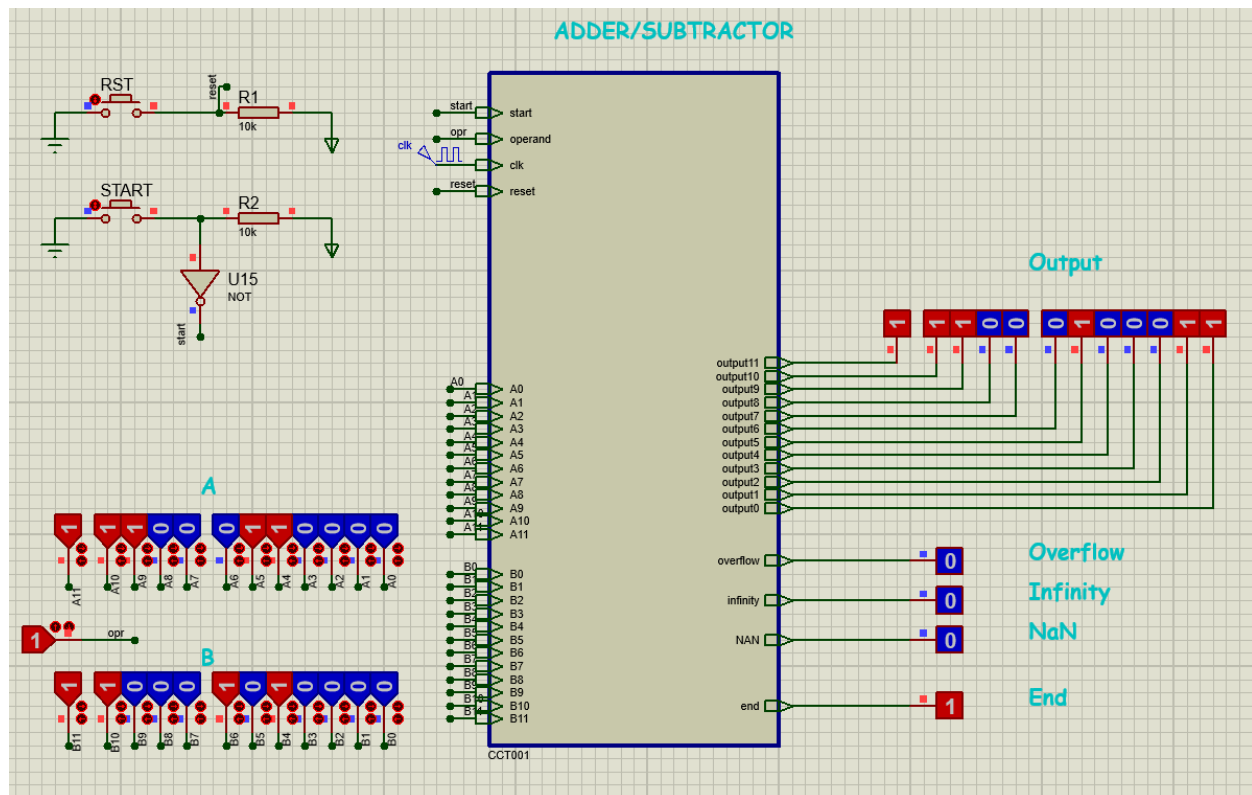
تست سوم: (تفریق دو عدد منفی)

$$A = 11100110000 = -44$$

$$B = 110001010000 = -3,25$$

$$\text{opr} = 1 \text{ (Subtract)}$$

$$\text{Output} = 11100100011 = -40,75$$



شکل ۱۸. تست سوم

همانطور که از مدار انتظار می‌رفت، خروجی مورد انتظار را مشخص کرده است.

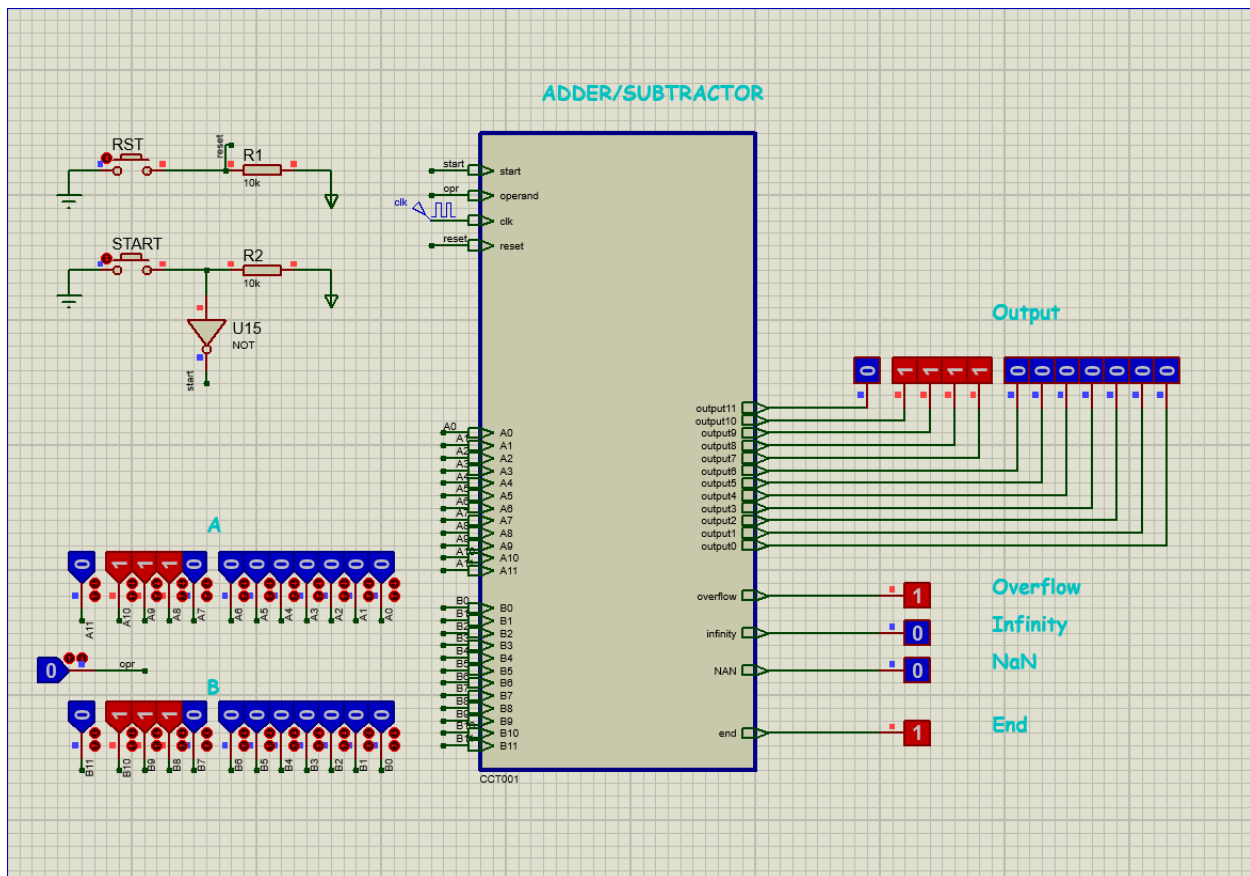
تست چهارم: (جمع منجر به اورفلو)

$$A = 011100000000 = 128$$

$$B = 011100000000 = 128$$

$$\text{opr} = 0 \text{ (Add)}$$

$$\text{Overflow} = 1$$



شکل ۱۹. تست چهارم

همانطور که از مدار انتظار می‌رفت، مقدار خروجی $\text{overflow} = 1$ را مشخص کرده است.

تست پنجم: (جمع با بی‌نهایت)

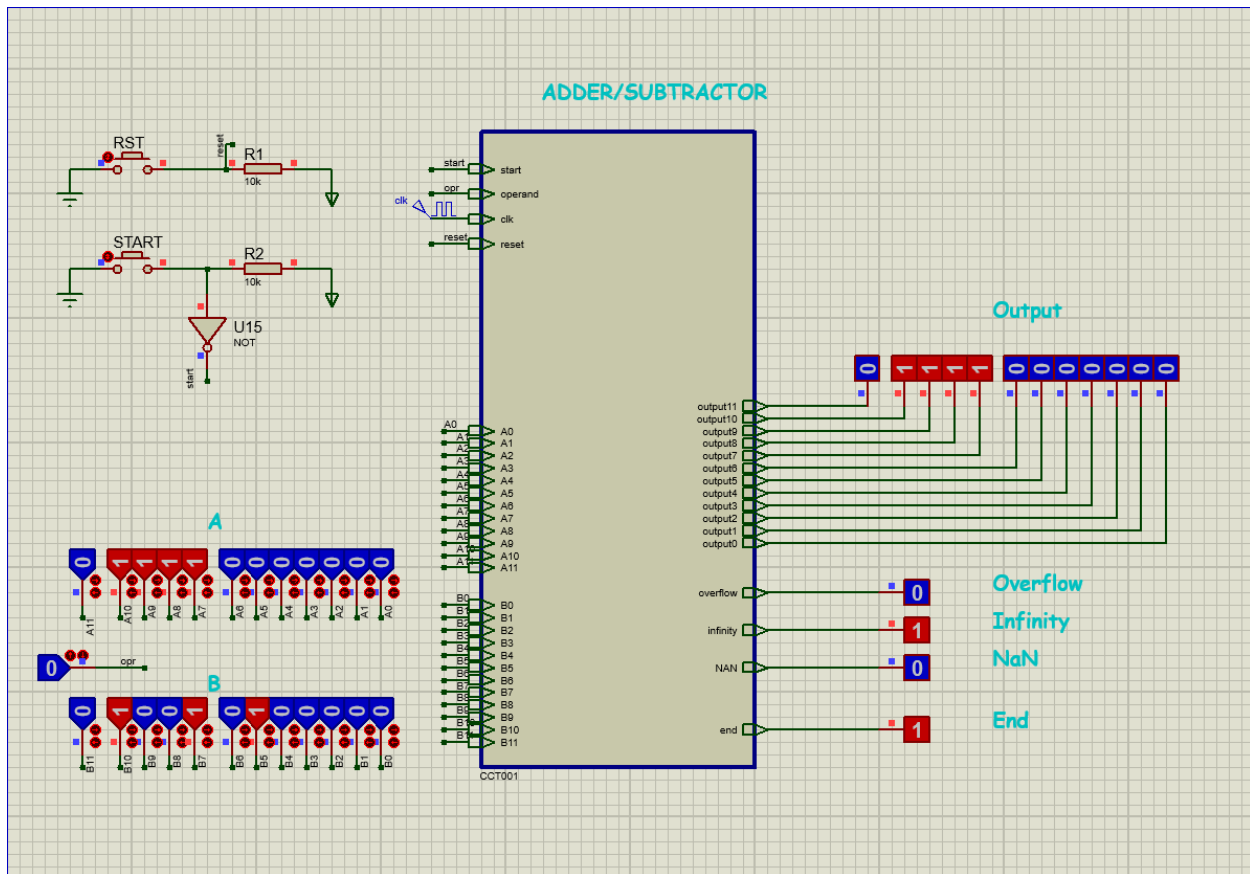
$$A = 0.1111 \dots = \infty$$

$$B = 0.1001 \dots = 0$$

$$\text{opr} = 0 \text{ (Add)}$$

$$\text{Infinity} = 1$$

$$\text{Output} = \infty$$



شکل ۲۰. تست پنجم

همانطور که از مدار انتظار می‌رفت، مقدار خروجی $\text{Infinity} = 1$ و $\text{Output} = \infty$ را مشخص کرده است.

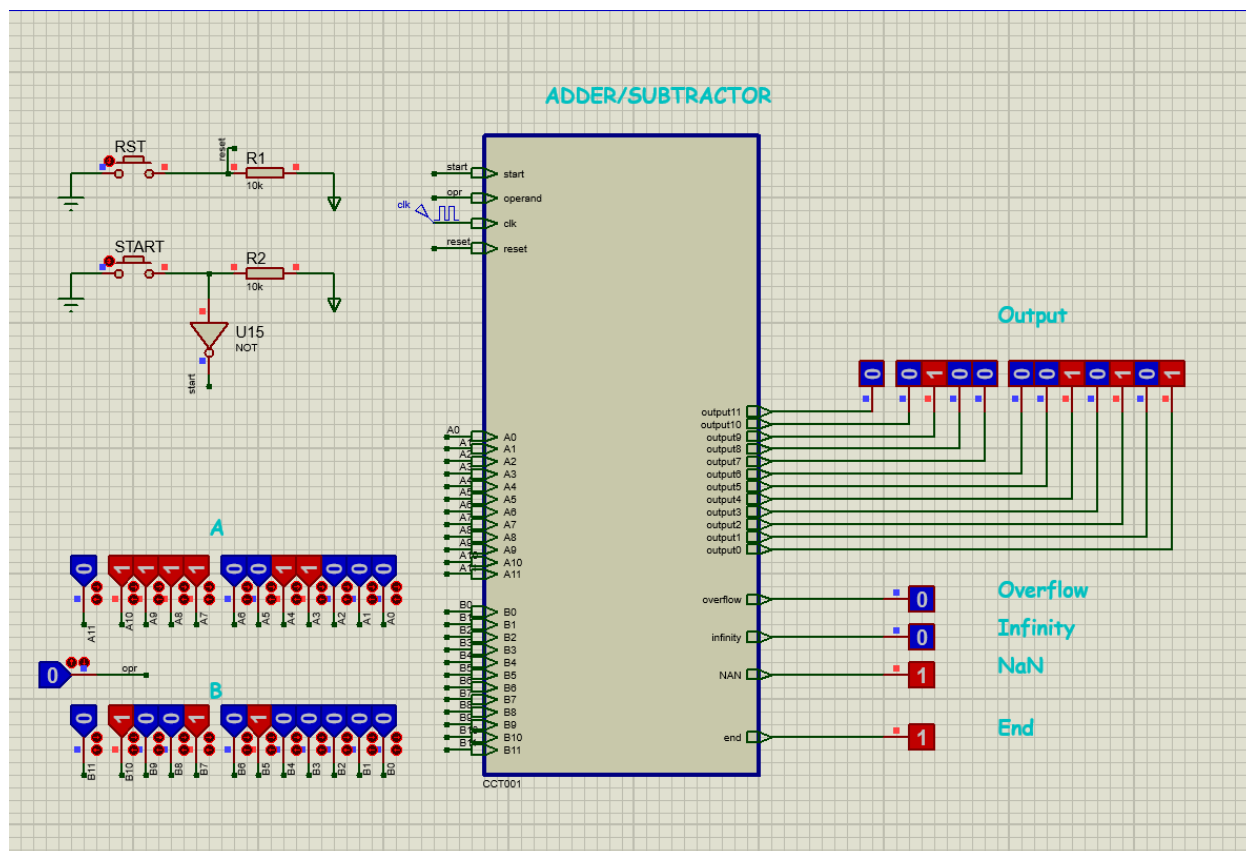
تست ششم: (جمع با NaN)

$$A = 0.1111011000 = \text{NaN}$$

$$B = 0.1001010000 = 0$$

$$\text{opr} = 0 \text{ (Add)}$$

$$\text{NaN} = 1$$



شکل ۲۱. تست ششم

همانطور که از مدار انتظار می‌رفت، مقدار خروجی $NaN = 1$ را مشخص کرده است.

تست هفتم: (جمع بی‌نهایت با NaN)

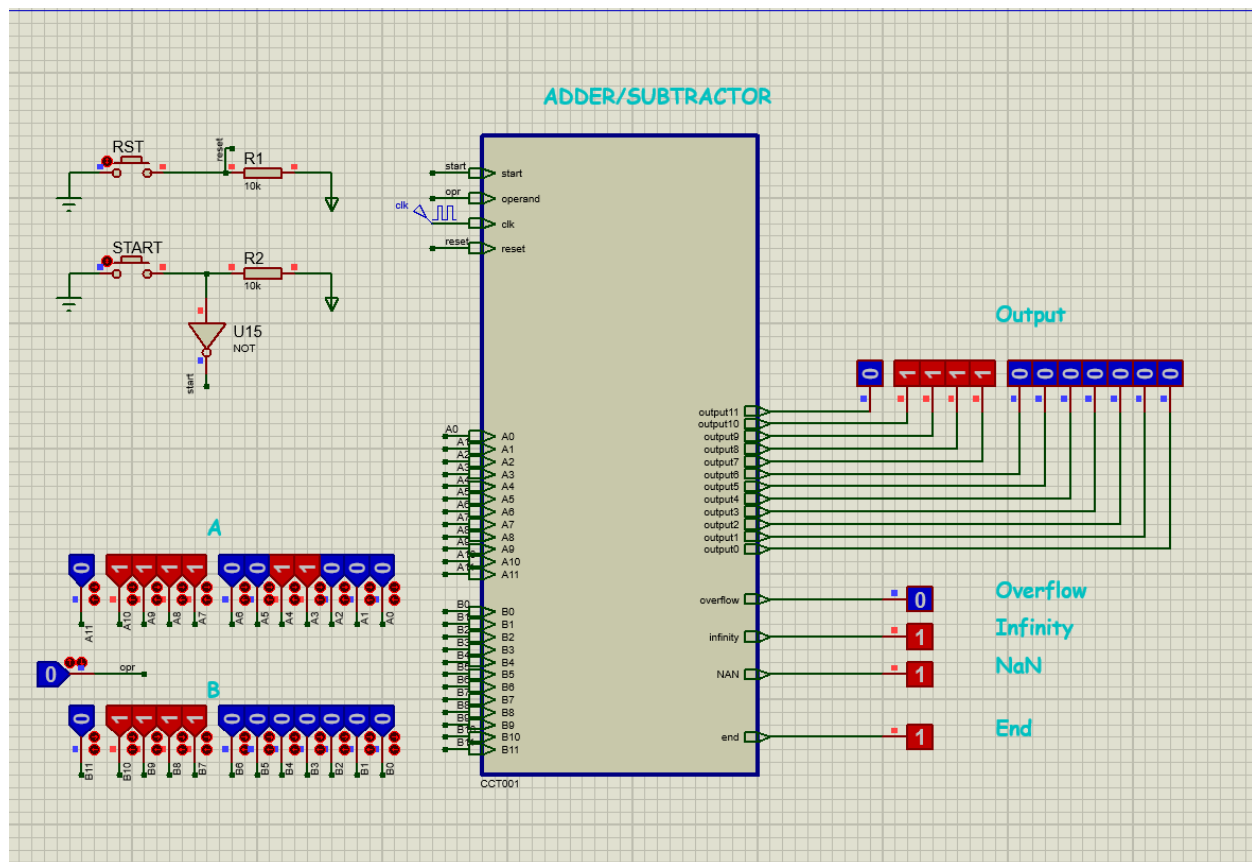
$$A = 011110110000 = NaN$$

$$B = 011110000000 = \infty$$

$$opr = 0 \text{ (Add)}$$

$$NaN = 1$$

$$Infinity = 1$$

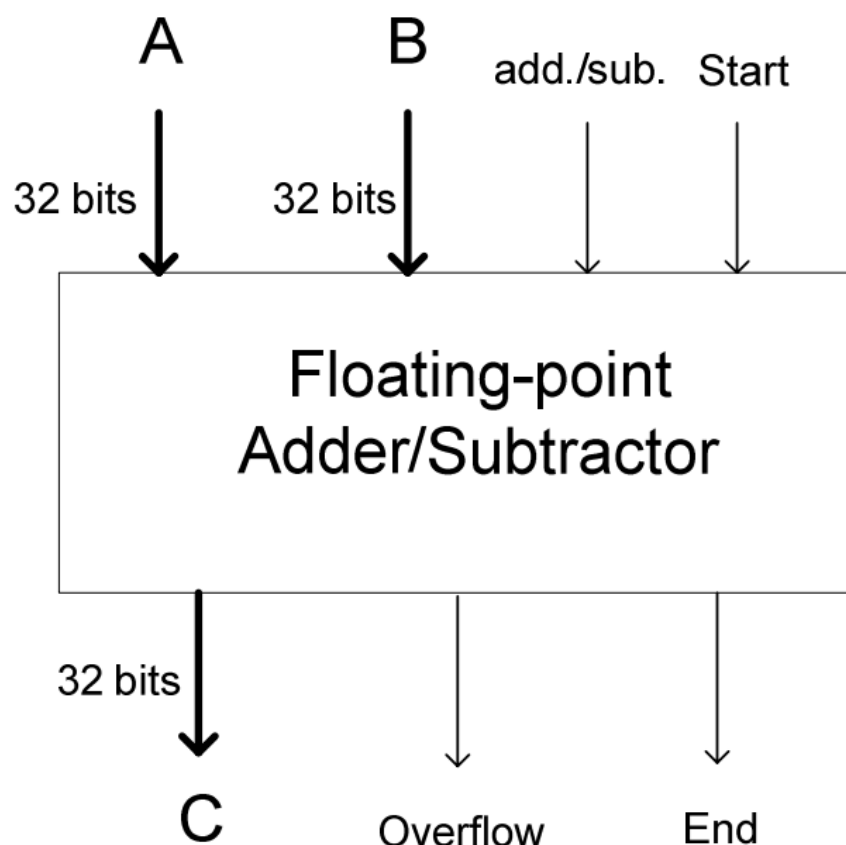


شکل ۲۲. تست چهارم

همانطور که از مدار انتظار می‌رفت، مقدار خروجی $\text{Infinity} = 1$ و $\text{NaN} = 1$ را مشخص کرده است.

نتیجه گیری:

در این آزمایش سعی کردیم با کمک گیت‌های موجود مانند شیفت رجیستر، جمع کننده و ... یک جمع/تفریق کننده اعداد ممیز شناور مطابق شکل زیر طراحی کنیم.



شکل ۲۳. طرح کلی مدار جمع/تفریق کننده ممیز شناور

مدار خواسته شده مطابق انتظار، طراحی و چندین تست نیز برای تعیین درستی مدار به آن داده شد که همه آنها را به درستی اجرا کرد و خروجی مورد نظر مشاهده شد.

منابع و مراجع:

- Mano, M. Morris. Computer system architecture. Prentice-Hall of India, ۲۰۰۳.
- Computer Organization & Design, The Hardware / Software Interface”, D. Patterson and J. L. Hennessy, Morgan Kaufmann Publishing, ۲۰۰۵.