برای طراحی این ماژول، ابتدا چند پارامتر اولیه تایین کردیم تا با استفاده از آنها بتوانیم طراحی خود را محدود به IEEE نکنیم و طراحی ما برای فرمت های دیگر هم ولید باشد ، اما به دلیل باگ هایی که خوردیم اینکار را نکردیم و از پارامتر ها صرفا برای مشخص کردن فرض هایمان استفاده کردیم. در ادامه سیم های input , output ,inout خود را نامگذاری کردیم که بلافاصله بعد از آن آمدیم و wire/reg های خودمان را تعریف کردیم. سپس شروع کردیم به مشخص کردن لاجیک سخت افزارمان و آن را بیشتر به طور رفتاری ساختیم.  
سخت افزار ما دارای 4 فلگ OF/UF/INF/NAN است که برای مشخص کردن حالت های خاص از آنها استفاده کردیم . البته میتوانستیم فلگ ZERO هم اضافه کنیم ولی چون از روی جواب میتوان صفر بودن را تشخیص داد ، این کار را نکردیم.  
ابتدا 2 ورودی داریم که میتوانیم آنهارا به سه بخش sign / exponent / fraction تقسیم کنیم. ابتدا آمدیم بخش های مختلف را از هم جدا کردیم تا کارمان ساده شود. برای پیدا کردن sign دو بخش ورودی را در هم xor کردیم و برای exponent هردو بخش را با هم جمع کردیم ، اما در اینجا مشکلی داشتیم آن هم مشکل وجود bias در فرمت IEEE بود ، برای همین مقدار 2n-1 – 1 را از این حاصل کم کردیم . برای قسمت fraction هر دو قسمت را در هم ضرب کردیم ، با اینکار یک عدد 47 بیتی به دست آوردیم(2 \* fraction\_bit\_count + 1) که به طور کلی با بیت های پرارزش تر آن کار داشتیم ، از F[47] برای نرمالایز کردن استفاده کردیم و از F[23:0] برای round کردن استفاده کردیم . در اخر با ملاحضه بیت round و بیت normalize قسمت مورد نیاز F را انتخاب کردیم و تغییرات را (در صورت نیاز) بر روی exponent نهایی انجام دادیم.

در اخر نیاز داشتیم که پاسخ را خروجی دهیم ، برای خروجی دادن پاسخ نیاز داشتیم که اول flag هایمان را چک کنیم ، اگر flag خاصی فعال نبود ، جواب عمومی را پس دادیم اما اگر فلگ های خاصی فعال بودند به صورت رو به رو عمل کردیم : اگر flag OF/UF فعال بودند x را پس دادیم چون جوابی برای نشان دادن نداشتیم ، اما اگر flag INF فعال بود ، عدد کدشده بینهایت را نشان دادیم و اگر flag NAN فعال بود ، عدد کد شده not a number را نشان دادیم .