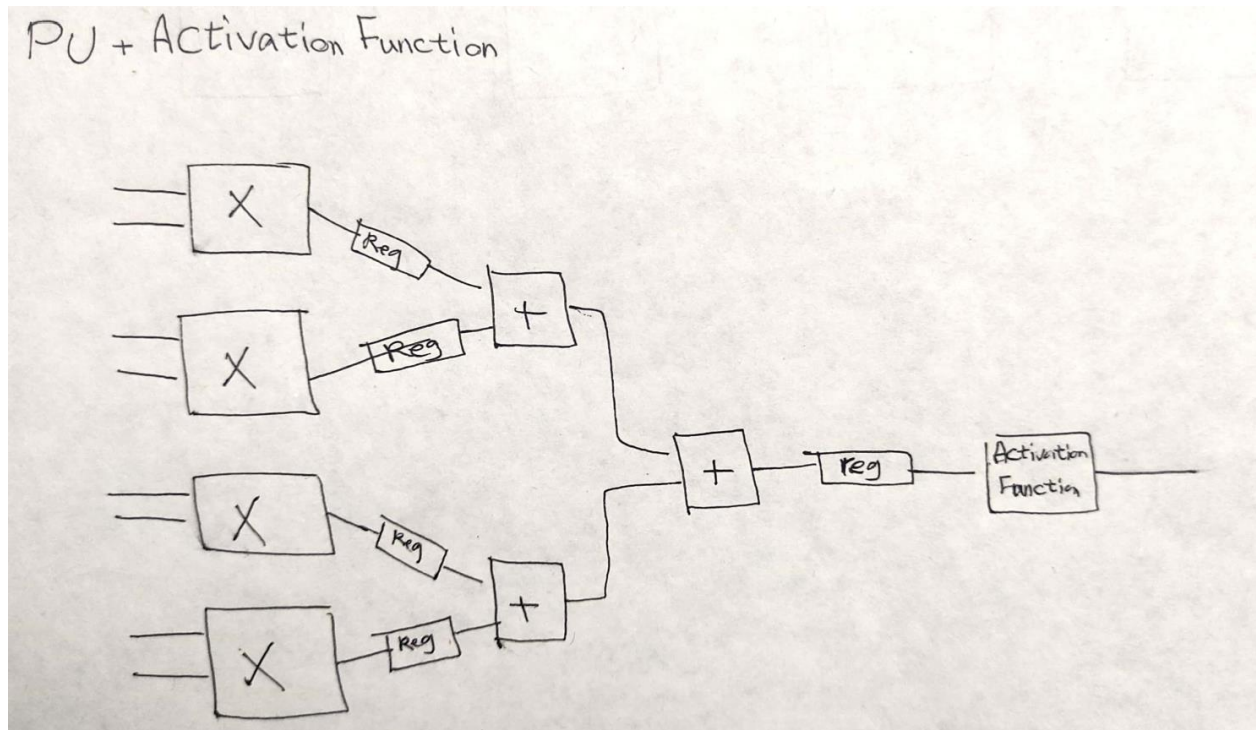


پروژه اول - طراحی کامپیوتری سیستم های دیجیتال

علی حمزه پور - 810100129

الهه خداوردی - 810100132

در پیاده سازی تابع فعال سازی را به داخل PU برده و آن ها را باهم قرار می دهیم.



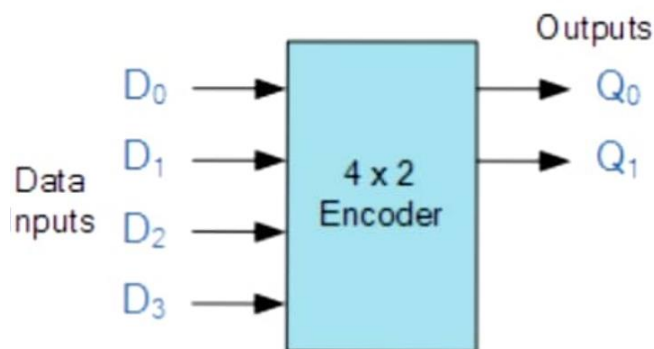
تابع فعال سازی را به این صورت تعریف کردیم که اگر بیت علامت عدد مورد نظر برابر یک بود یعنی عدد منفی است و در این صورت با یک multiplexer به ازای بیت علامت یک عدد صفر را انتخاب می کنیم.

در مرحله بعد برای چک کردن اتمام مراحل و پیدا کردن بزرگترین مقدار در حالی که سه عدد میان چهار عدد صفر شده، در هر عدد همه بیت هارا باهم or کرده و جواب آن را برای 4 عدد باهم به صورت یک 4 wire بیتی درآورده و چک میکنیم که آیا عدد برابر 0001 یا 0010 یا 0100 یا 1000 است یا خیر.

پیاده سازی این قسمت از آنچه در فاز اول گفته شد متفاوت است.

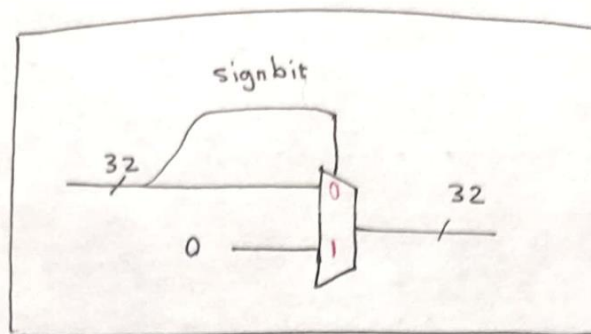
```
assign is_finished = (a == 4'b0001 || a == 4'b0010 || a == 4'b0100 || a == 4'b1000);
```

برای قسمت خروجی دادن بزرگترین عدد نیز از یک multiplexer استفاده کردیم. برای مقداردهی سلکتور این multiplexer همان 4 رقمی که در قسمت چک کردن اتمام داشتیم را به encoder داده و سپس پاسخ encoder را سلکتور multiplexer قرار می دهیم.



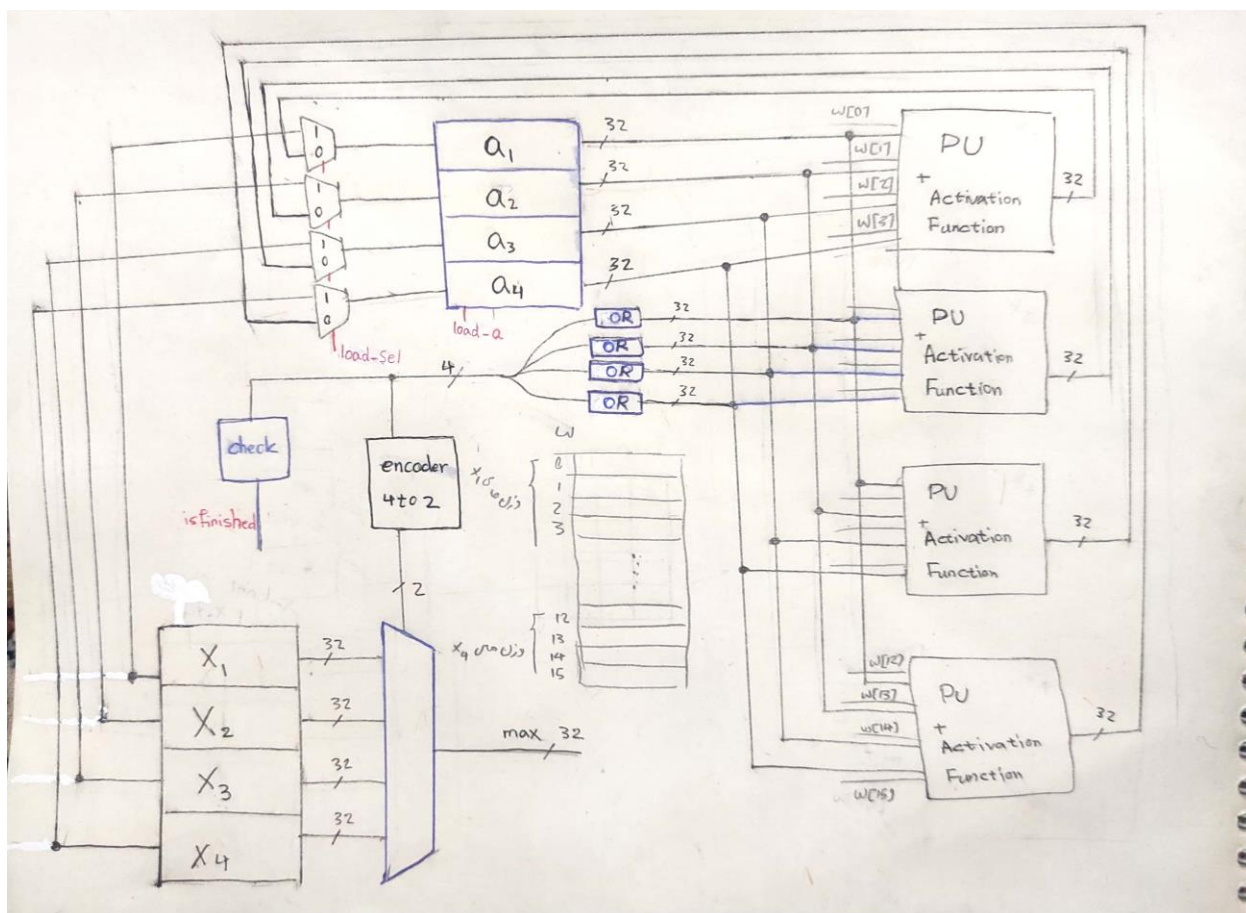
Inputs				Outputs	
D ₃	D ₂	D ₁	D ₀	Q ₁	Q ₀
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1
0	0	0	0	x	x

Activation Function



طراحی datapath

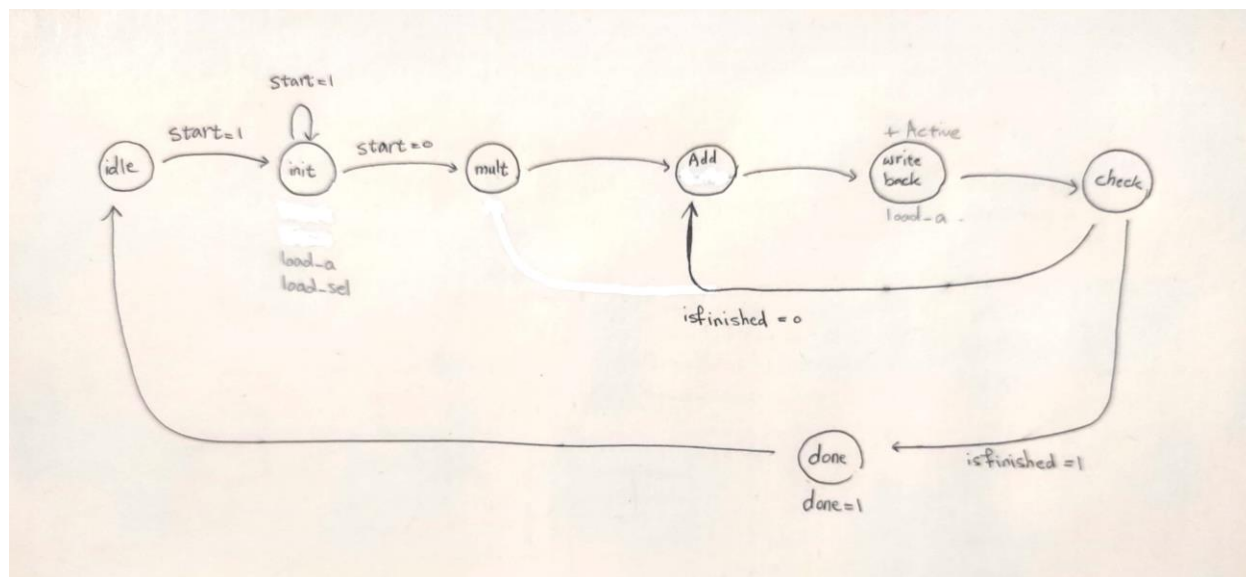
به دلیل تعداد زیاد wire های مربوط به W و واضح بودن نحوه اتصال آن‌ها از کشیدنشان به صورت مستقیم خودداری شده تا نتیجه واضح تر باشد، در عوض برای مثال و مشخص شدن روند تعدادی از آن‌ها نوشته شده‌اند.



تغییر نسبت به آنچه گفته شد نحوه ذخیره سازی W هاست.

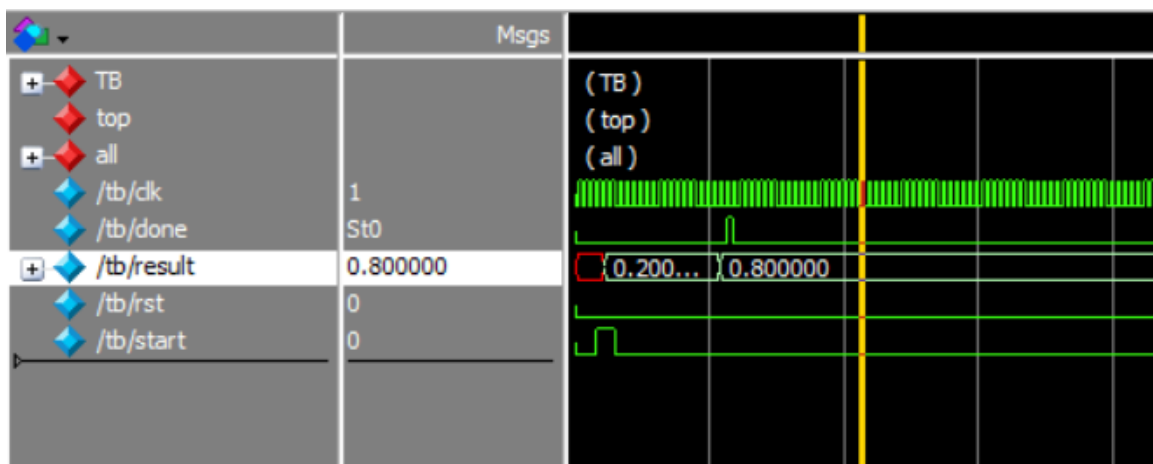
Controller

تغییر کنترلر نیز نسبت به قبل حذف سیگنال های init است.



تست

ابتدا اعداد 0.2، 0.4، 0.6، 0.8 را به عنوان تست اجرا می کنیم.



سپس ورودی های 0.231، 0.75، 1.497، 1.136 را امتحان می کنیم.

