

به نام خدا

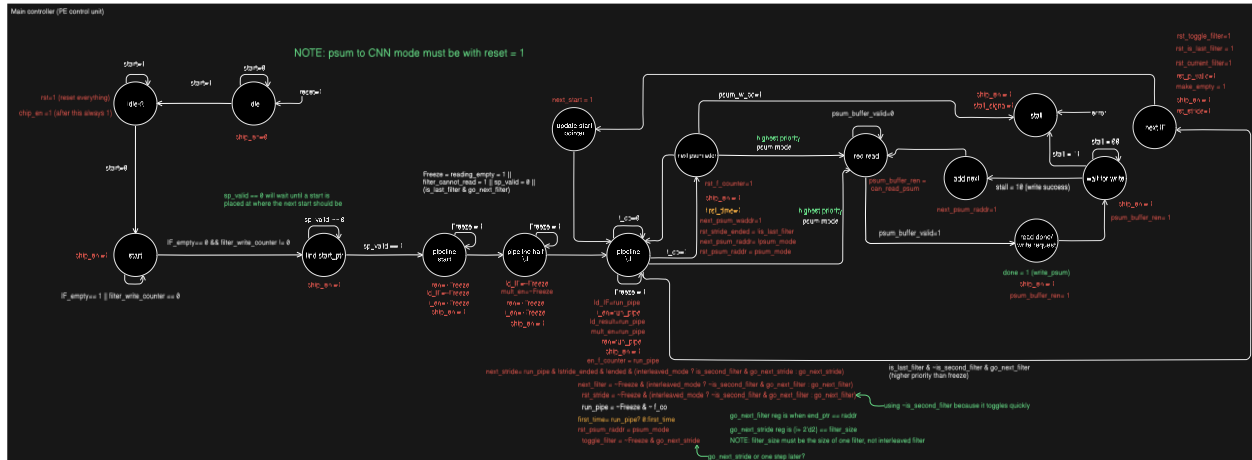
پروژه 5 فاز 2 CAD

آریان رجبی 810199421

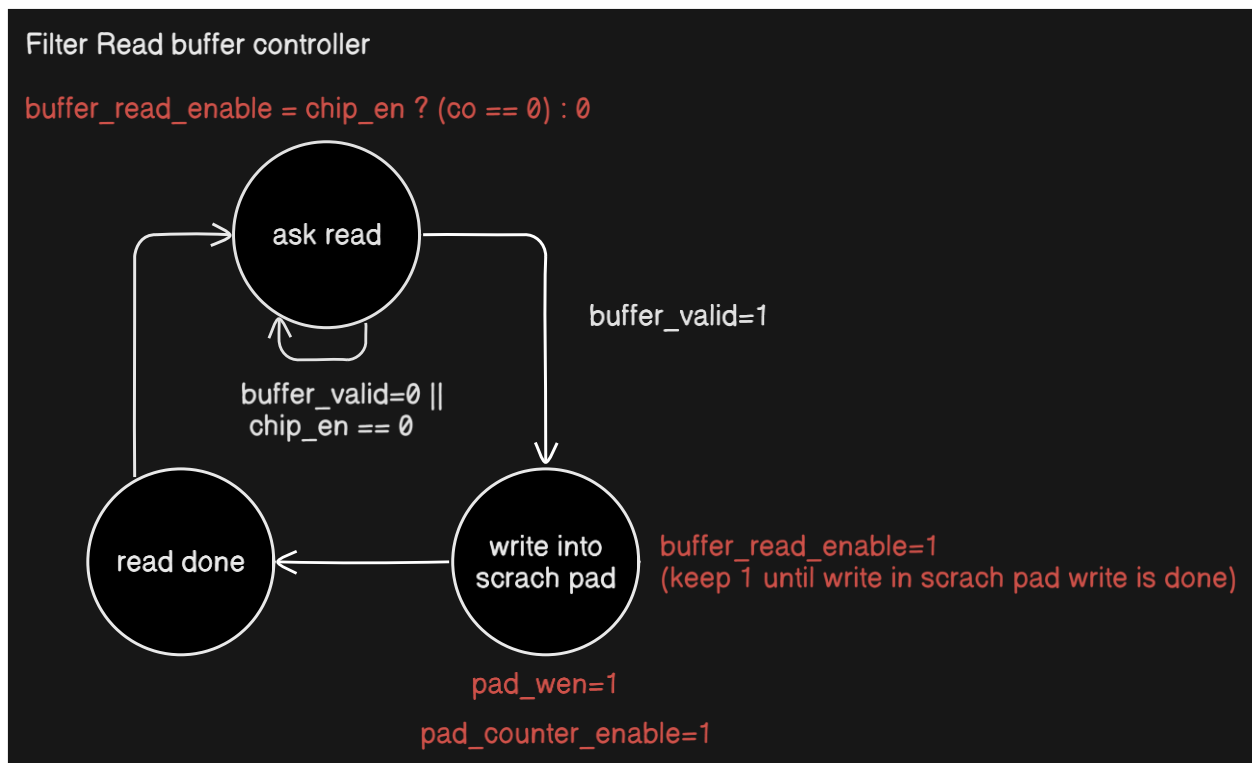
پریا پاسبان 810101393

توجه: فایل با کیفیت svg کنار PDF قرار داده شده

کنترلر اصلی:

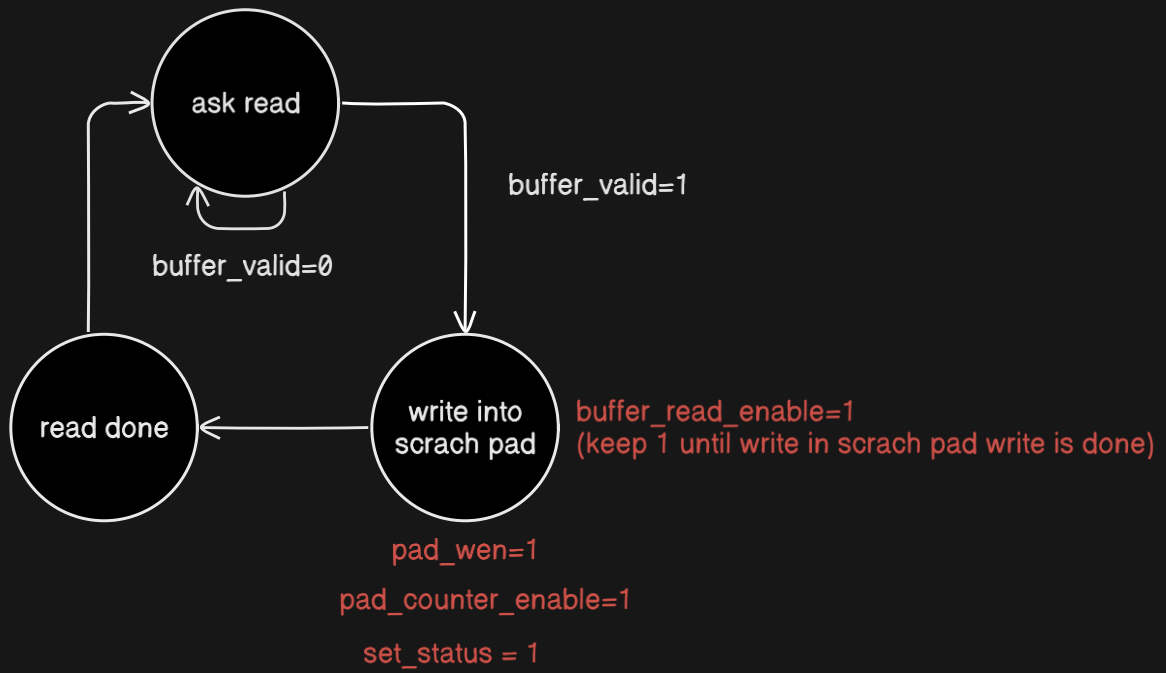


کنترلر خواندن از بافر ها:



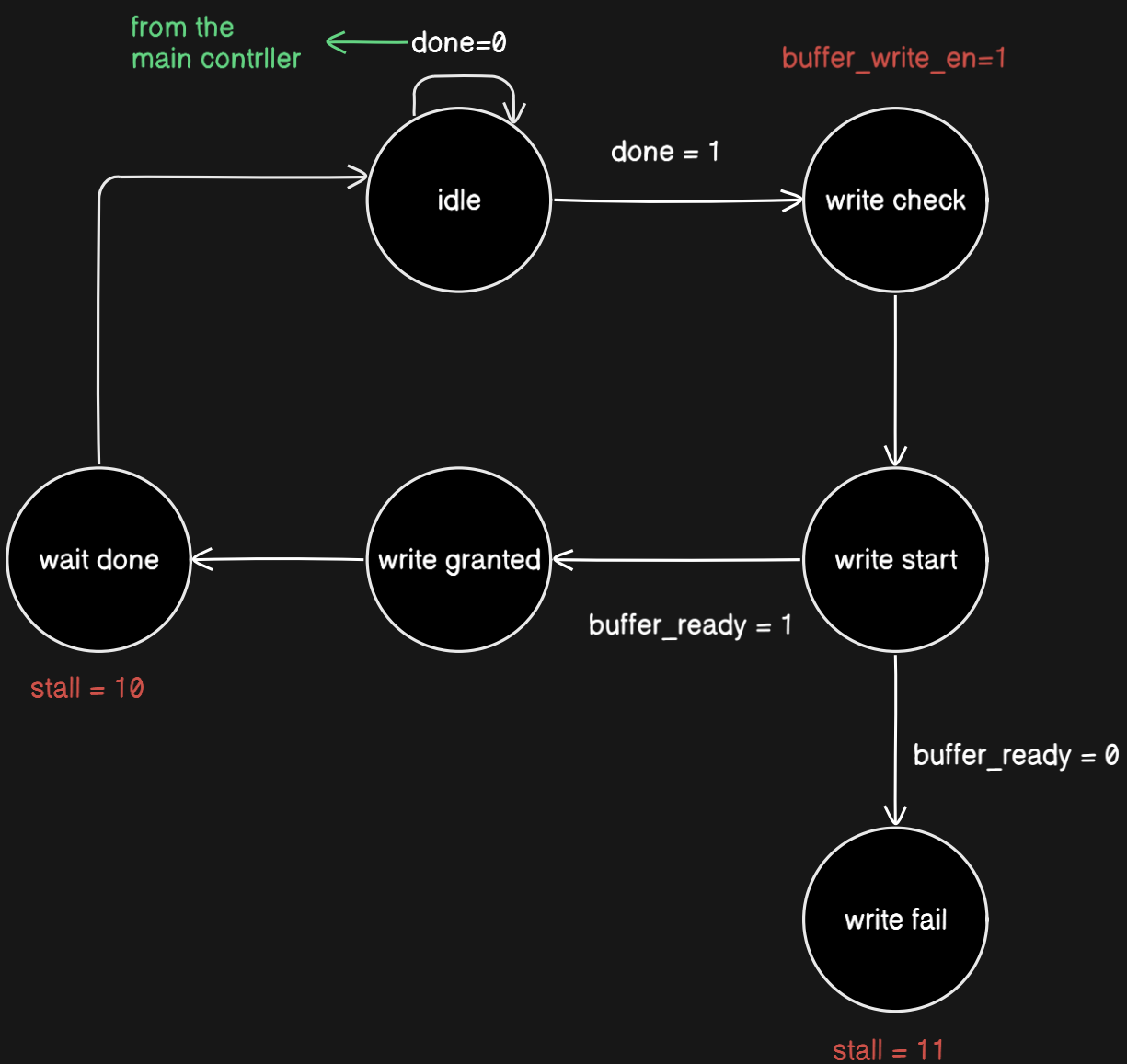
IF Read bufffer controller

`buffer_read_enable = (IFMap_can_write == 1)`

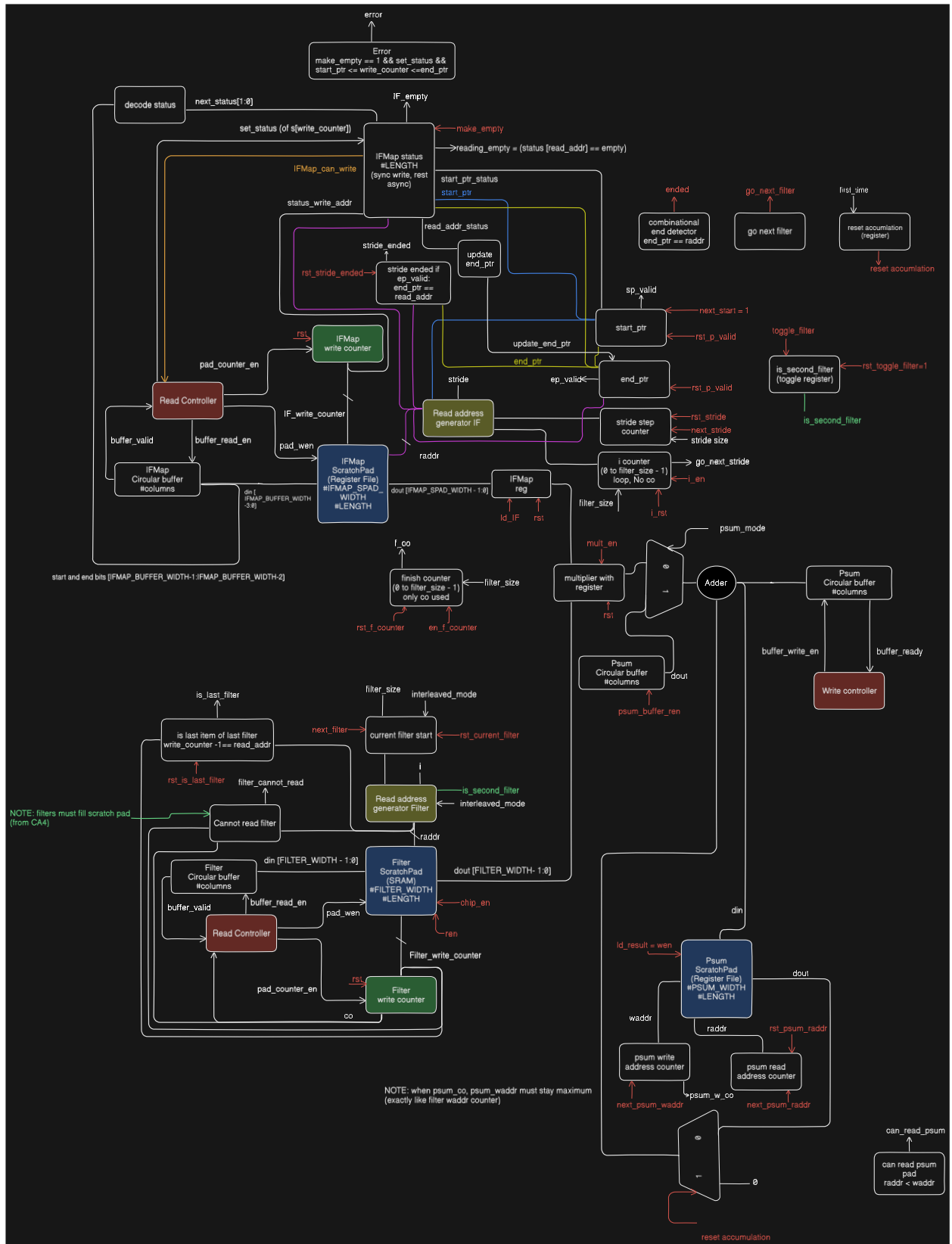


نوشتن در بافر خروجی:

Write buffer controller



دیتا پس:



- در پروژه 4، ما فرض کرده بودیم سایز فیلتر از 3 بزرگ تر است که نصف تست کیس های این پروژه به همین دلیل امکان اجرا نداشتند (با این حال برای فیلتر های بزرگ تر کاملاً درست کار می کند)
- در تست کیس های ما، خروجی display شده در کنسول، خروجی psum مود است که هر خروجی با عددی جمع شده (به ترتیب 1 تا n) یعنی بافر ورودی psum برای جمع:

1
2
3
4
5
6
7
8
9
10
11
12

- است، پس از اعداد display شده باید موارد بالا را کم کرد تا حاصل psum کانولوشن به درستی به دست بیاید.
- در پروژه 4 اورفلو stall نمی شد و ادامه می یافت، برای همین در پروژه ما هم استال نمی شود (البته پارامتر ها را به گونه ای داده ایم که اصلاً اورفلو نکند) برای همین، 8 خروجی تولیدی در مود 1، 2 تای اول مطابق تست کیس هستند، 2 تای دیگر ولی نه (به خاطر اورفلو) مقدار محاسبه شده توسط کد ما درست است چون اورفلو ندارد.
 - یک سری فرضیات و نکات با رنگ سبز روی کنترلر یا دیتاپس نوشته شده اند.
 - سناریو 1 و 3، محاسبات به روش همان پروژه 4 هستند، با این تفاوت که در مود 3 فیلتر سایز جمع سایز فیلتر ها است، به همین منظور، ما این دو مورد را یکسان در نظر گرفته ایم و کد به جای مود، یک interleaved mode باینری می گیرد که نشان گر تفاوت سناریو 2 با بقیه است. سایز فیلتر در سناریو 1 و 3 اید به درستی تنظیم شود که در تست کیس ها انجام شده.
 - کلیت پیچیدگی مسئله روی کنترلر است به همین دلیل تعداد استیت زیاد شده، همچنین دیتاپس ماژول های زیادی می بینیم که کمک کننده adres generator ها هستند و می توانستیم برای سادگی آن ها را نیاریم.

تغییرات نسبت به فاز یک:

کلیات یکسان، جزییات نیاز به تغییر زیاد به دلیل interleaved mode