

به نام خدا

کنترلر و دیتاپس پروژه دوم طراحی کامپیوتری سیستم های دیجیتال

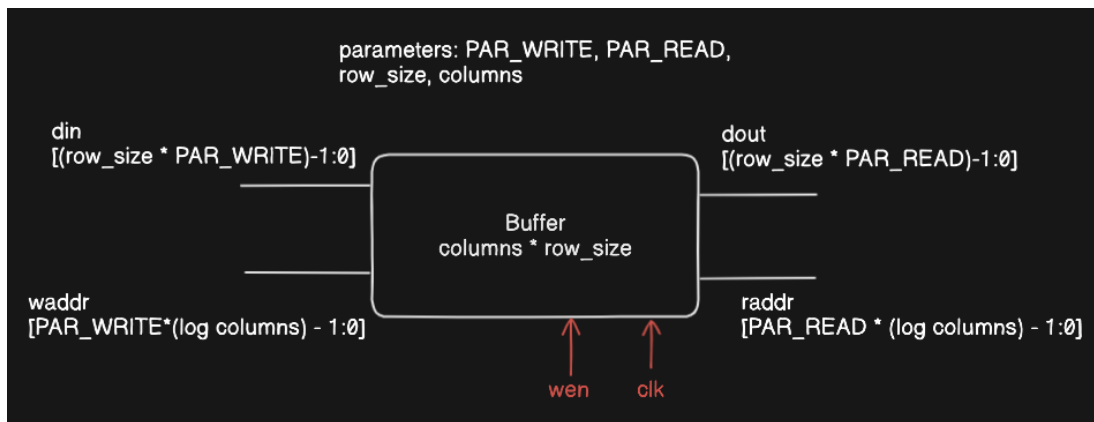
آریان رجی 810199421

پریا پاسه ورز 810101393

توجه: تغییرات کنترلر در فاز 2 در صفحه 4 ذکر شده اند.

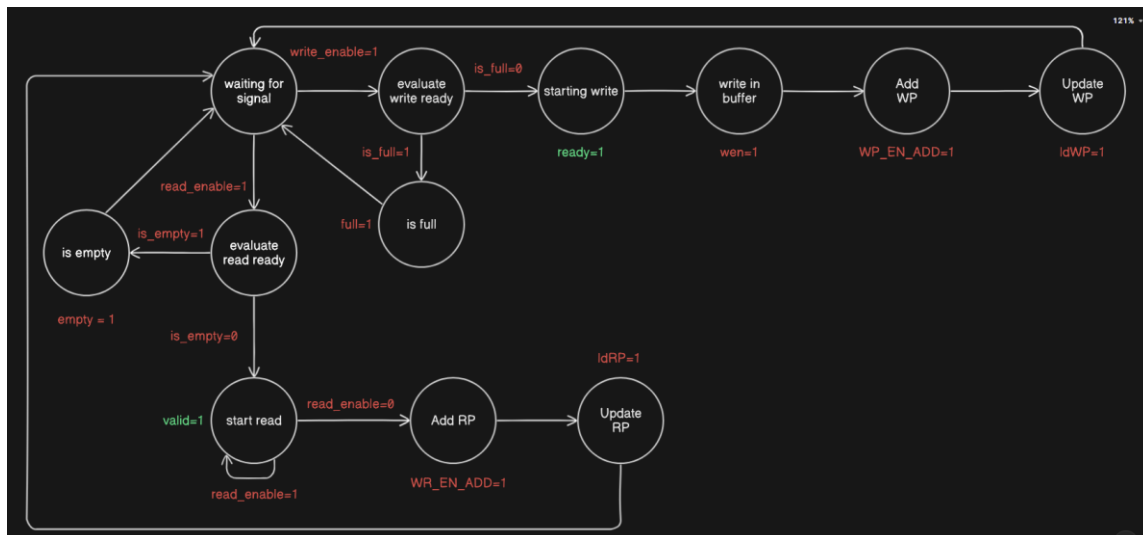
ماژول بافر خطی:

- لگاریتم ها بر مبنای 2 هستند و سقف بالای آن ها در نظر گرفته خواهد شد. (اگر وریلاگ برای پارامترها لگاریتم نداشته باشد، باید به صورت دستی وارد شود)



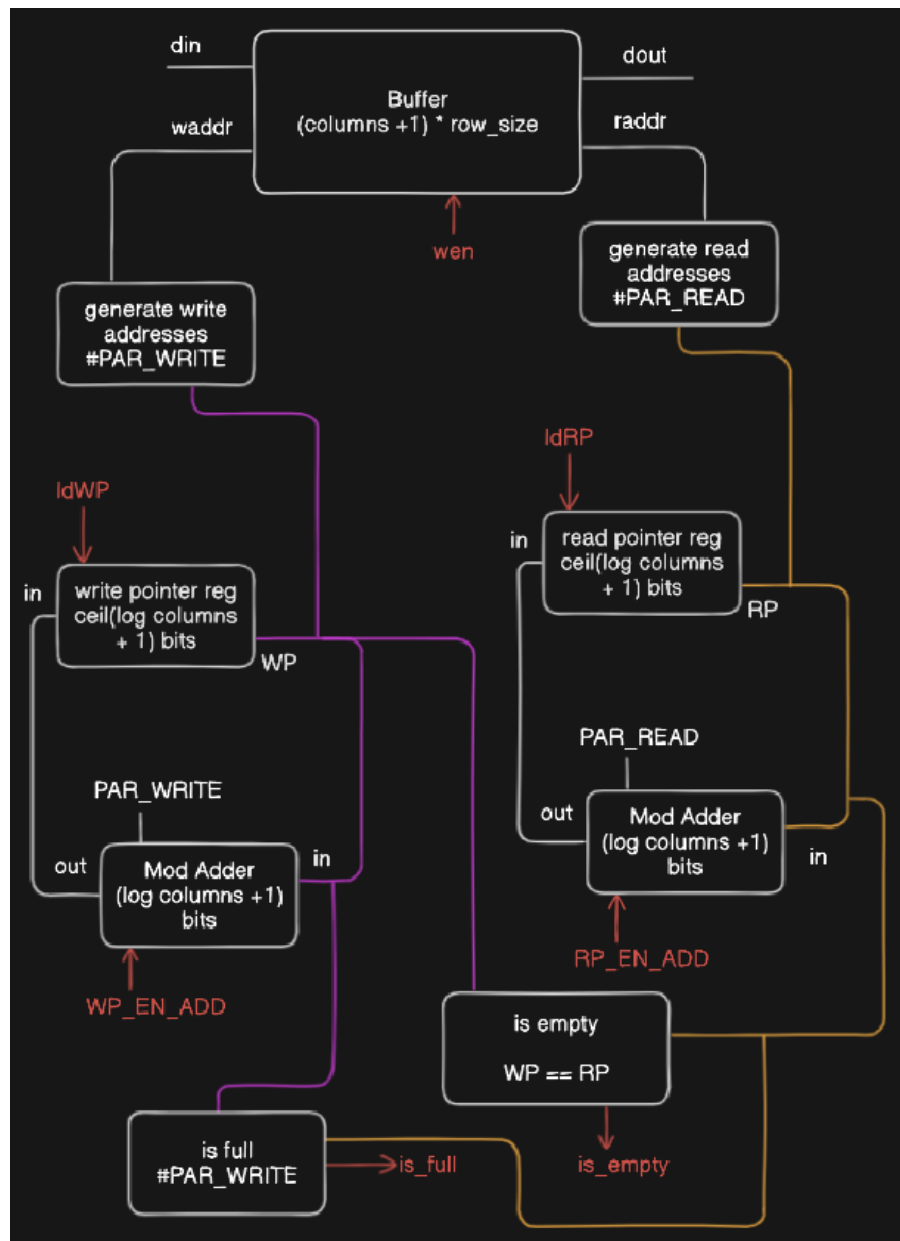
کنترلر بافر حلقوی:

- کاربر بیرونی، باید `read_enable` را در طول خواندن 1 نگه دارد و بعد اتمام آن، 0 کند.
- سیگنال های `empty` و `full` فقط وقتی معتبر هستند (صادر می شوند) که کاربر درخواست خواندن یا نوشتن دهد، این دو سیگنال به اندازه 1 سایکل خواهند بود.

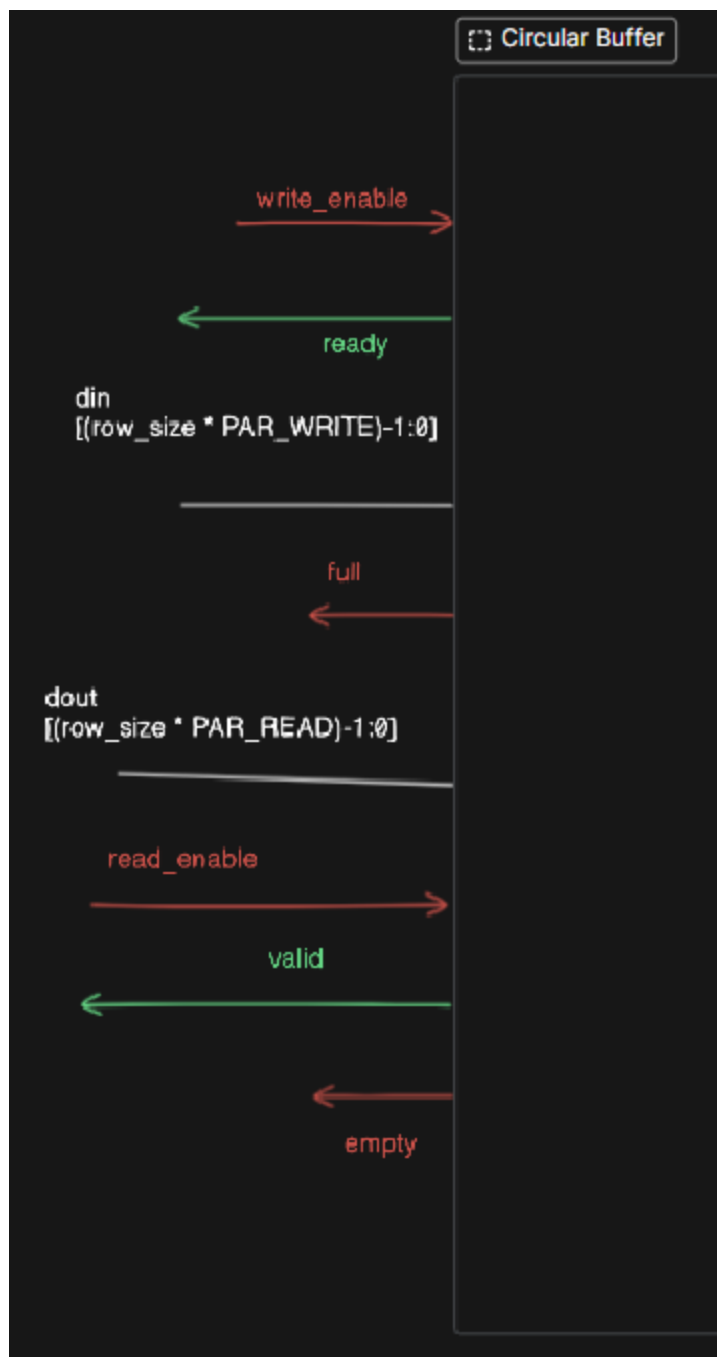


دیتاپس:

- بافر خطی داخلی، باید یکی بیشتر از بافر حلقوی اصلی باشد تا در حالتی که بافر کاملاً خالص است (WP==RP)، PAR_WRITE برابر تعداد کل سائز بافر باشد، به درستی بافر کامل پر شود. (اشتباه در فاز 2 تصحیح شده، توضیحات صفحه 4)
- پارامترهای PAR_READ و PAR_WRITE باید کمتر مساوی اندازه بافر باشند.
- ماژول Mod Adder در داخل خود از % استفاده نخواهد کرد. (چون سنتز آن سخت است).
- ماژولهای generate read/write addresses و is full با این که مثل عمل mod می کنند، اما باز از % استفاده خواهند کرد.

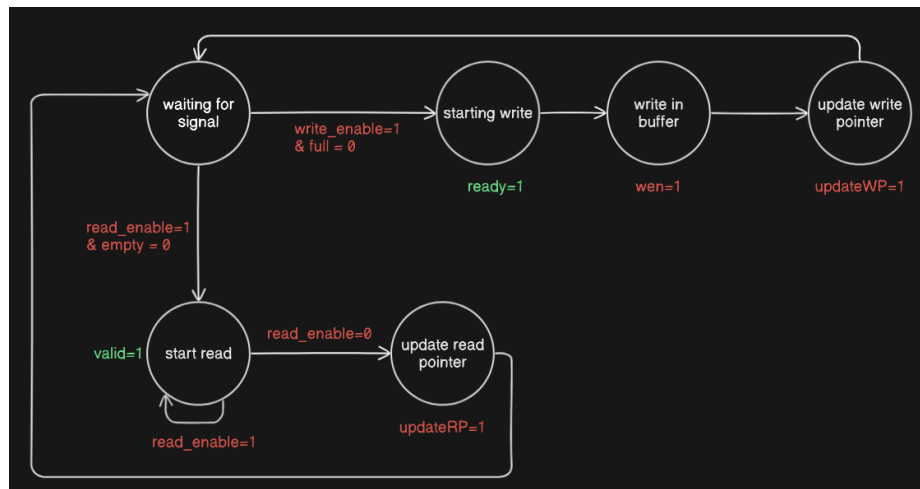


به صورت کلی، ماژول بافر حلقه ای داخل خود کنترلر دارد و سیگنال و ورودی های زیر را دریافت و صادر می کند:



تغییرات کنترلر و دیتاپس نهایی (تغییرات کنترلر فقط برای کاهش state ها بودند):

- آپدیت کردن write pointer و سپس ریختن در رجیستر مربوطه به یک state منتقل شده اند (مشابه read pointer) - برای کاهش تعداد state ها
- State های evaluate ready و is full حذف شده اند، سیگنال فول به یه قطعه combinational تبدیل شده و همیشه در خروجی ماژول وجود خواهد داشت (قبلا یک کلاک صادر میشد) (مشابه برای evaluate read ready و is empty)



- ماژول های mod adder و read register ادغام شده اند (به خاطر تغییر اول کنترلر)، مشابه برای write هم همین موضوع را داریم.
- منطق خالی بودن بافر یا پر بودن آن اشتباه بود و $WP == RP$ ناقص است، در ماژول های مربوط به فول و خالی، منطق درست قرار گرفته است.

