به نام خدا

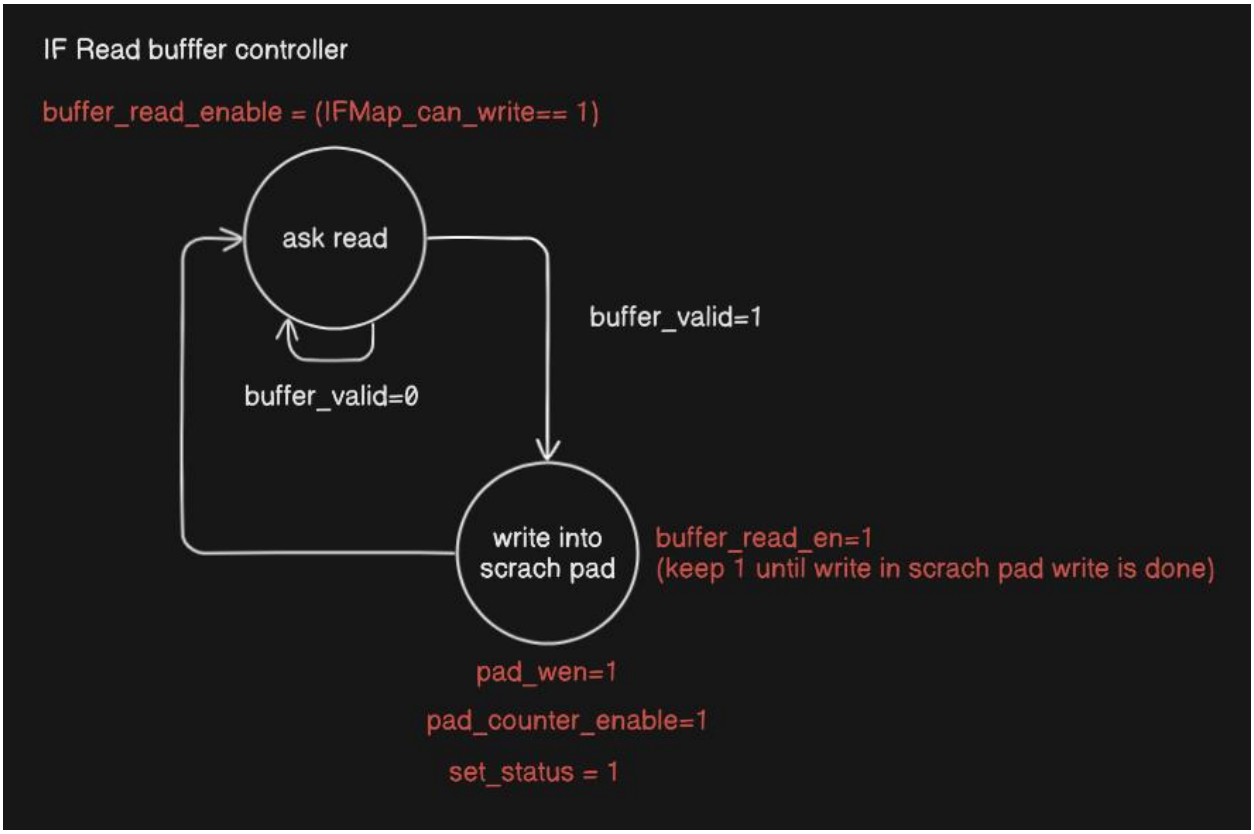
گزارش کار فاز اول تمرین کامپیوتری 4 درس CAD

آریان رجبی 810199421 پریا پاسه ورز 810101393

کنترلر خواندن از بافر IF:

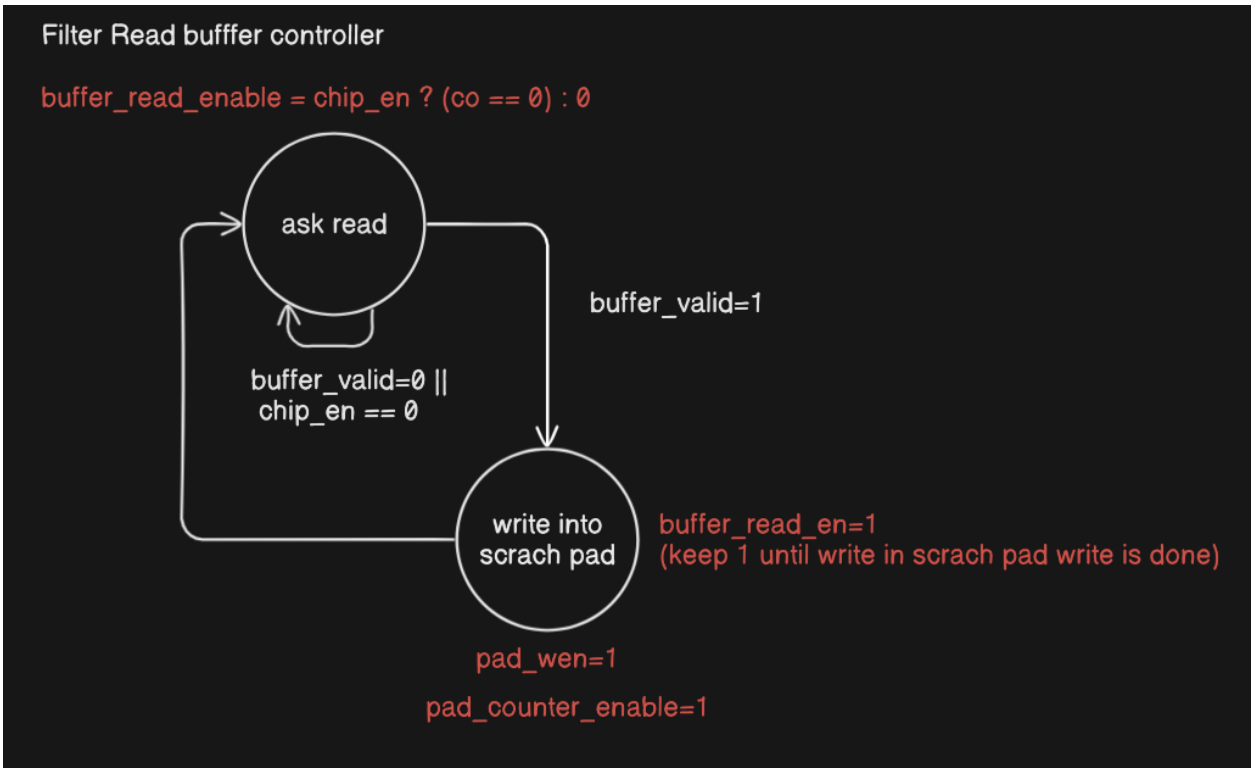
هر موقع امکان نوشتن در اسکرچ پد باشد، این کنترلر نوشتن را شروع میکند.

(PAR_READ = 1)



کنترلر خواندن از بافر فیلتر:

سیگنال chip_en اضافه شده و کمی سیگنال ها تغییر کرده اند.

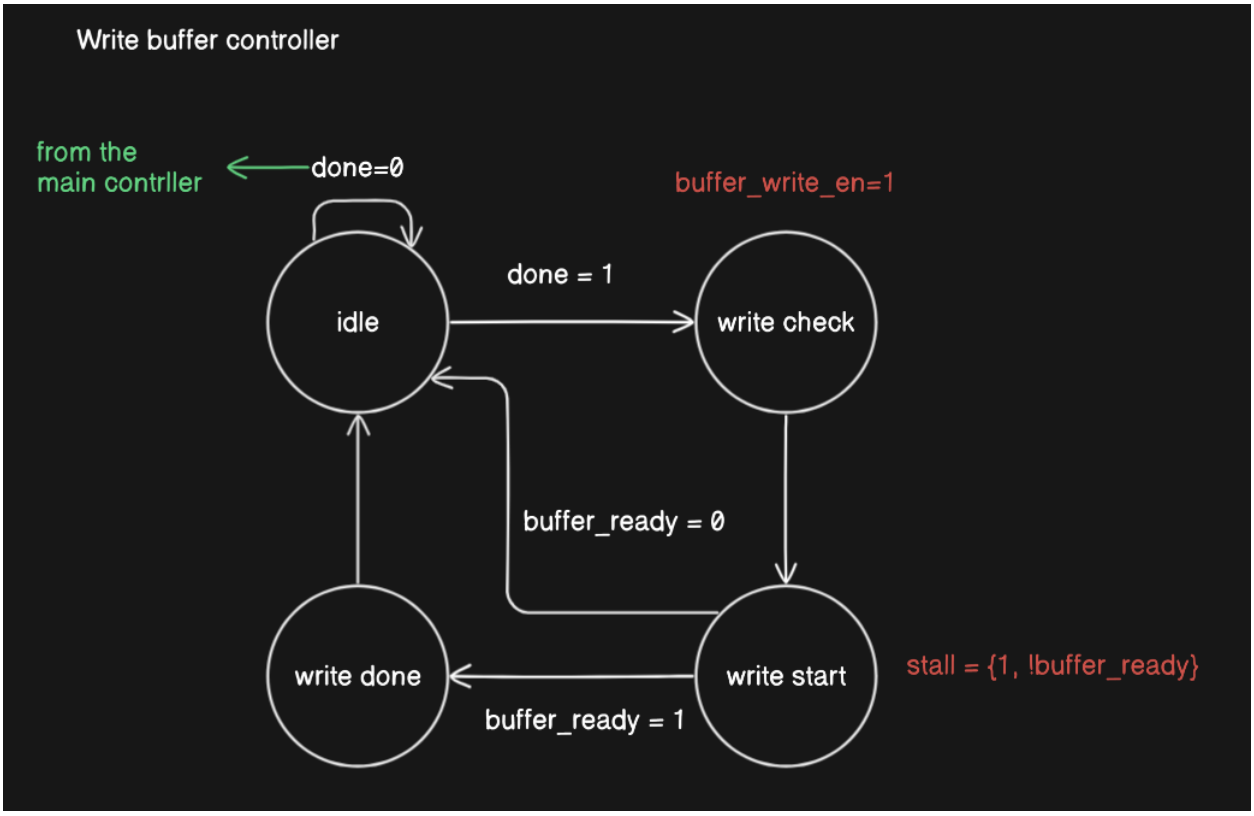


کنترلر نوشتن در بافر خروجی:

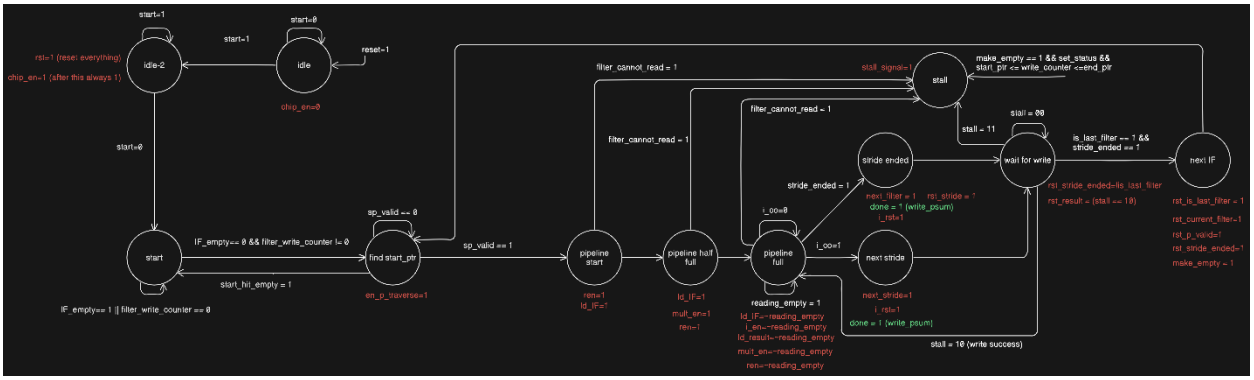
PAR_WRITE = 1 است، خواندن از این بافر ربطی به این پروژه ندارد و میتوانیم در تست بنچ PAR_READ را عدد دلخواه قرار دهیم.

stall is a 2bits signal where:

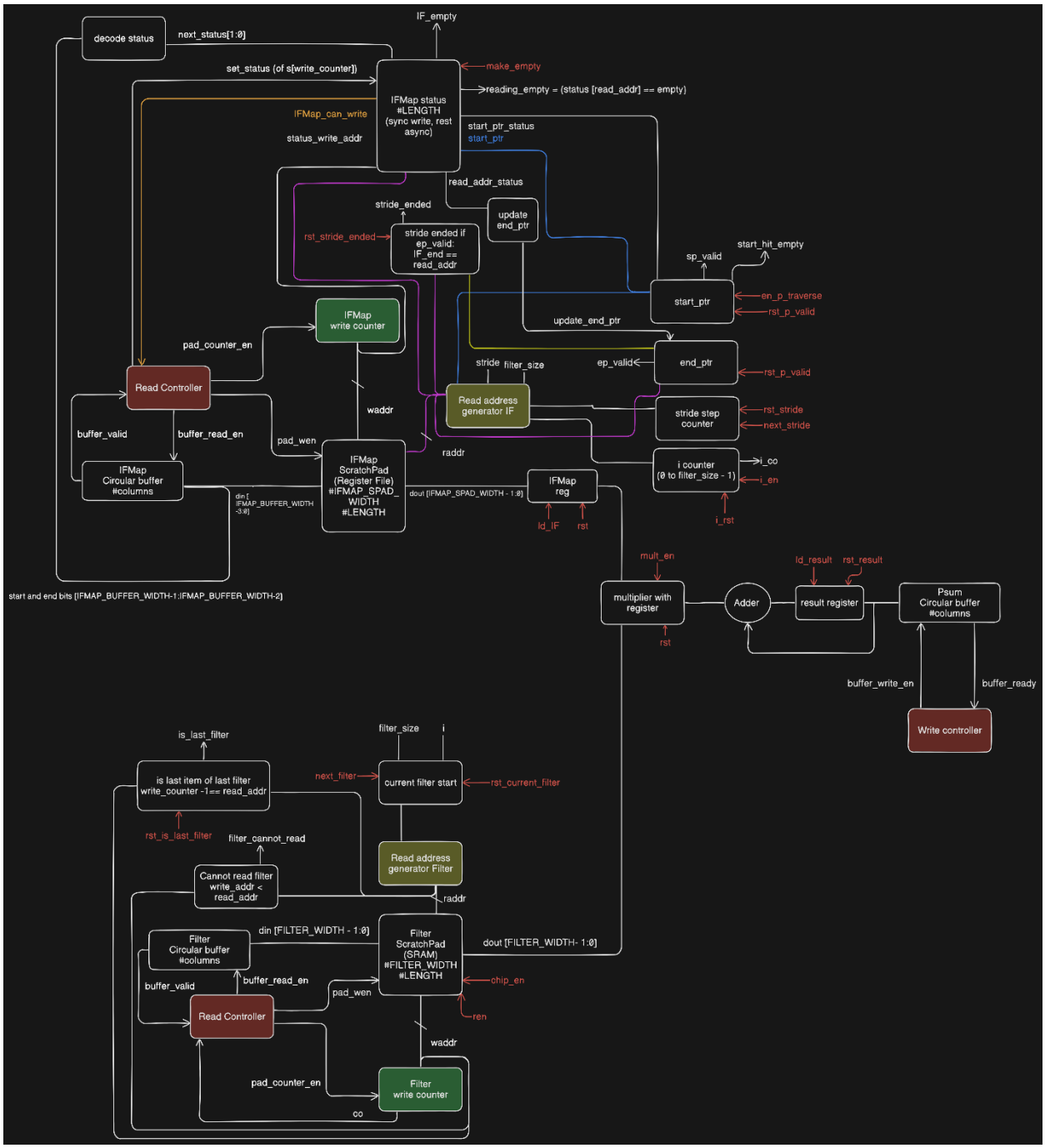
- 00 -> stall not valid
- 01 -> never happens
- 11 -> stall
- 10 -> write success


Write buffer controller
from the main contrller
done=0
buffer_write_en=1
idle
done = 1
write check
buffer_ready = 0
write done
write start
stall = {1, !buffer_ready}
buffer_ready = 1

کنترلر اصلی:

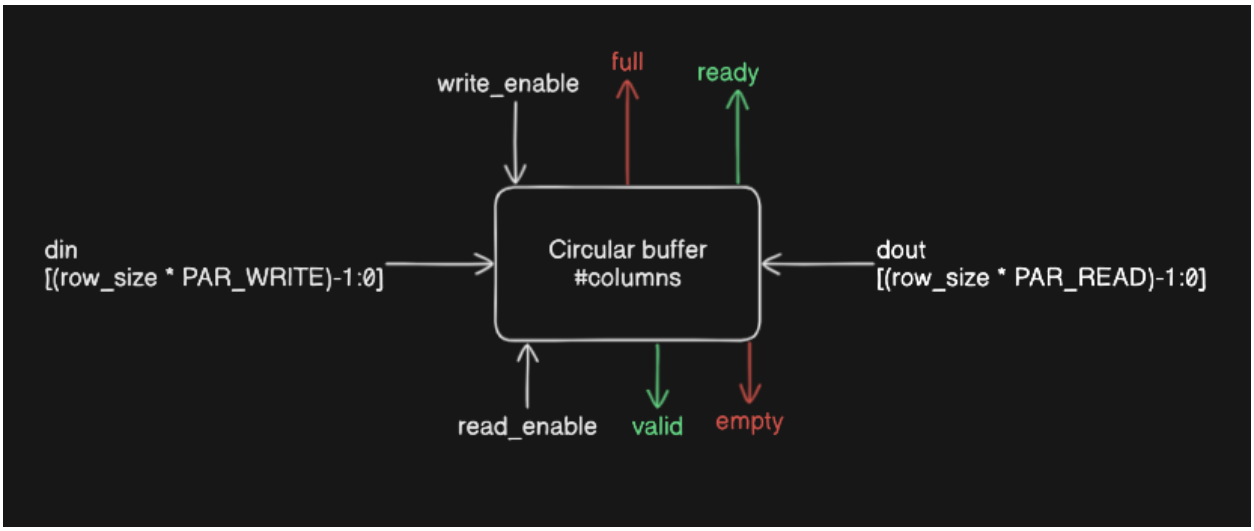

idle-2
idle
start
find start_ptr
pipeline start
pipeline half full
pipeline full
next stride
stall
stride ended
wait for write
next IF
start=1
start=0
reset=1
chip_en=0
sp_valid == 1
start_hit_empty = 1
en_p_traverse=1
ren=1
ld_IFs=1
mult_en=1
reading_empty = 1
stall = 10 (write success)
stall = 11
stall = 00
stride_ended = 1
i_cc=0
i_cc=1
next_stride=1
done = 1 (write_psum)

دیتاپس:



توجه: کیفیت بالاتر تمام تصاویر گزارش در فایل svg کنار PDF قرار گرفته اند.

از بافر حلقه ای تمرین کامپیوتری خودمان استفاده کرده ایم:



توجه: در جلسه توجیهی ذکر شد که حداکثر 2 تا IF در اسکرچ پد قرار می گیرند، اما در دیتاپس ما محدودیتی برای این موضوع در نظر گرفته نشده، و به کمک مموری IFmap status وضعیت هر درایه IF را ذخیره می کنیم (هماهنگ شده با TA)

نکات:

1- there's no need to check the buffer empty in "read buffer controller" because it is internally checked in the buffer itself (same for buffer full)

2- Write buffer controller knows that it has to wait one more cycle to make sure the buffer write is completed (write done state is added due to lack of 2 way handshake in circular buffer)

3- if map status:

00 -> empty (DEFAULT)

01 -> start (is the start of an IF)

10-> end

11 -> None (between start and end)

4- IFMap can write = (IFMap_status[write_counter] == empty)

5- IF_empty = if all bits of IFmap status table are 0

6- IFMap read address generator (given these inputs, address can easily be generated with + and *):

address = f(IF_start_addr,filter_size,stride, stride_step, i)

7- filter_cannot_read has the highest precedence in all 3 pipeline states

precedence in pipeline full:

- reading_empty
- stride_ended
- i_co

precedence in wait for start:

- stall = 00
- stall = 11
- is_last_filter (here stall is inevitably 10 because of precedence)
- stall = 10

in find start & end pointers, start_hit_empty has precedence

8- filter read address generator:

address = f(current_filter_start, i)

9- Scratch pad is a linear memory of LENGTH items each with a size of WIDTH

10- assume PAR_READ is 1, because scratch pad can write one at a time

11-IFmap buffer #row_size = IFMAP_BUFFER_WIDTH

12- is_last_filter once issued, can only be reset (will keep value of 1 until reset)

13- sp_valid and ep_valid stay 1 until rst

14- calculations on IF scratchpad addr (read and write) are circular, but in filter, co after done (no replacement)

15- start_hit_empty = (IFstatus [sp] == empty) (stay 1 until reset)

16- stall is a 2bits signal where:

- 00 -> stall not valid
- 01 -> never happens
- 11 -> stall
- 10 -> write success

17- stride_ended stay 1 until reset

18- filter_size = 1 could cause problems! (in is_last_filter)

19- make_empty makes the values in range start_ptr and end_ptr empty in IF status table

20- some signals and wires may be missing

21- in IFMap status table, if make_empty == 1 && set_status == 1 && start_ptr <= write_counter <=end_ptr, the controller goes to stall as an error, even though this should not be possible at all, because set status from read controller is meant to set the status of an empty item in scratchpad

22- stall means error, can exit stall with a reset signal

23- chip_en is 1 right from the beginning to the end, because after start, there's always possibility of read or write to the filters

24- start_ptr is sync, end_ptr is asnyc (combinational)

25- update_end_ptr -> if read_addr_status is 10 (end) set ep_valid = 1 and update end_ptr value to read_addr

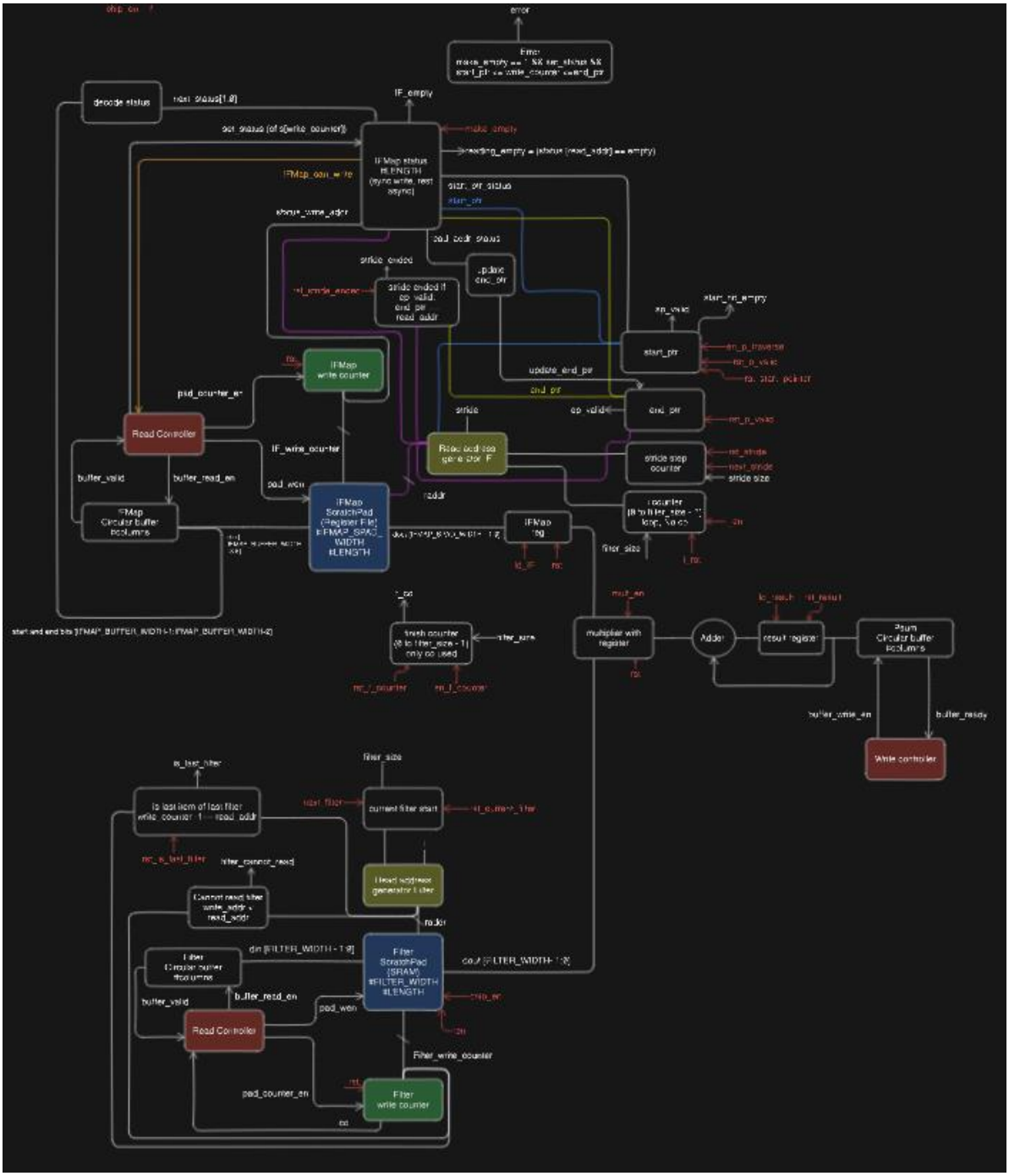
26- in pipeline_full state, if curently reading empty, everything will freeze, and as soon as reading_empty is 1, in the next clk (because controller is sync) it'll resume the work

به نام خدا

گزارش کار فاز دوم تمرین کامپیوتری 4 درس CAD

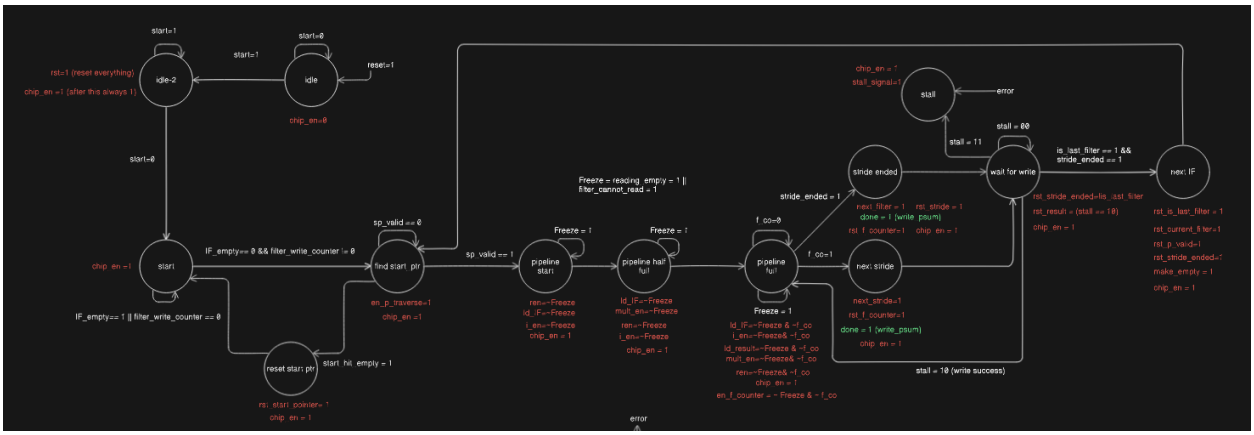
آریان رجبی 810199421 پریا پاسه ورز 810101393

تغییرات دیتاپث:

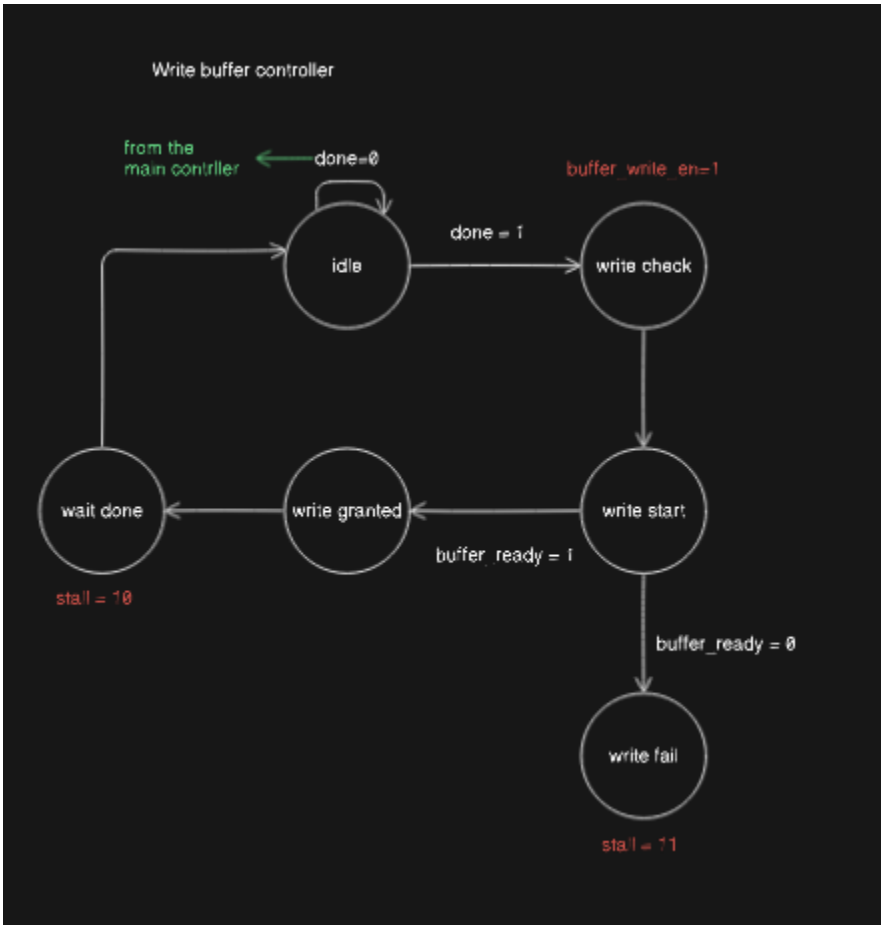


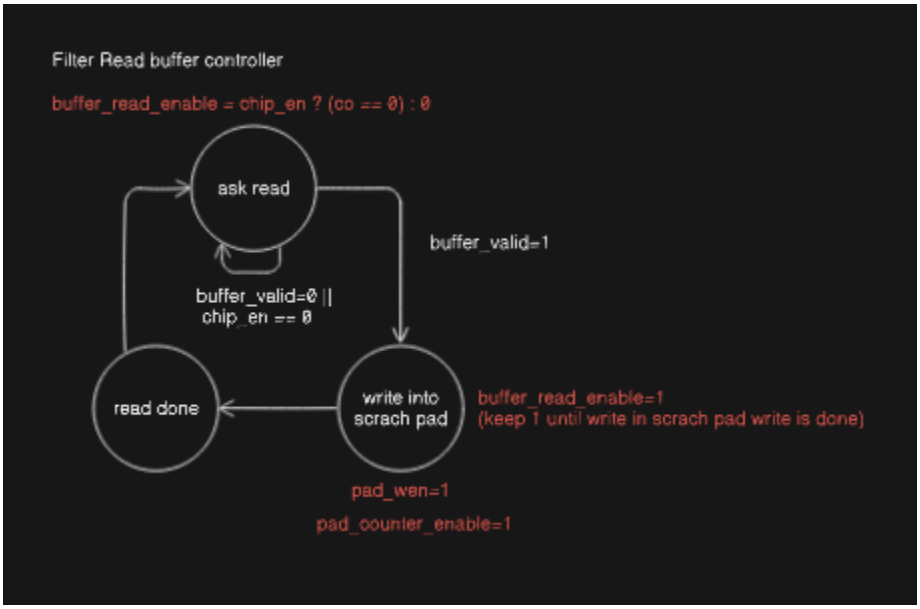
تغییرات کنترلر:

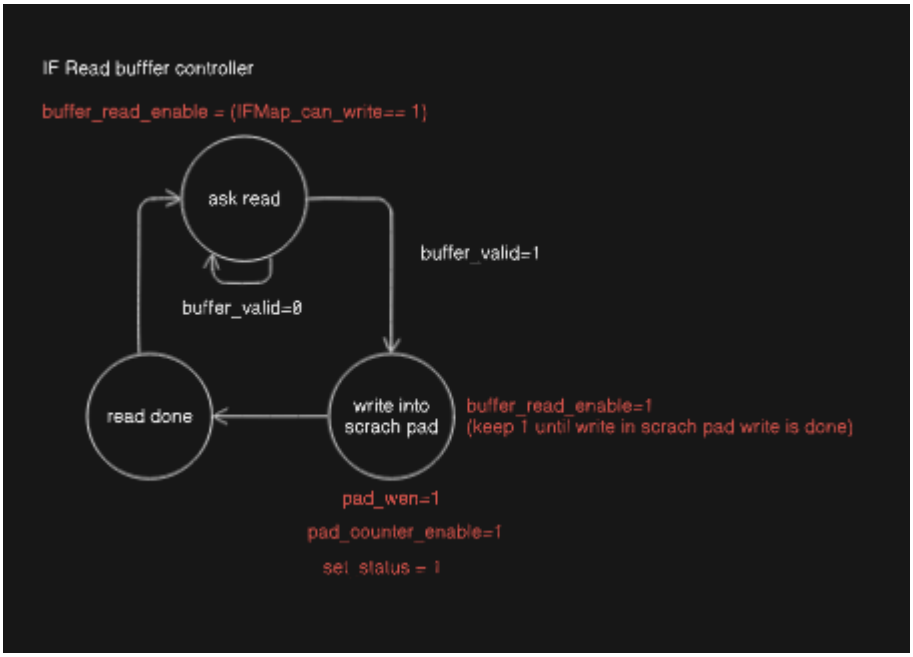
کنترلر اصلی:



سایر کنترلرها:




Filter Read buffer controller
buffer_read_enable = chip_en ? (co == 0) : 0
ask read
buffer_valid=1
buffer_valid=0 ||
chip_en == 0
read done
write into
scrach pad
buffer_read_enable=1
(keep 1 until write in scrach pad write is done)
pad_wen=1
pad_counter_enable=1


IF Read bufffer controller
buffer_read_enable = (IFMap_can_write== 1)
ask read
buffer_valid=1
buffer_valid=0
read done
write into
scrach pad
buffer_read_enable=1
(keep 1 until write in scrach pad write is done)
pad_wen=1
pad_counter_enable=1
set status = 1