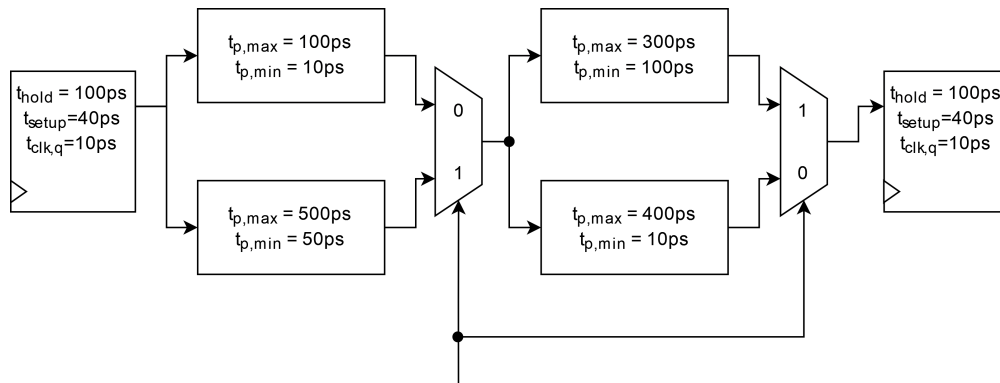# EECS 151/251A: Homework 9

Due Friday, April 29th

Problem 1: The evil, timing violation...

Consider the following sequential circuit, with the timing characteristics of the flip-flops and combinational blocks annotated. Assume that the muxes have no propagation delay, and that their inputs have been stabilized.

Part a)

Assuming no clock skew and no clock jitter, what is the minimum clock period of this design? Show your work.

> To begin, we ignore any hold violations since we are just doing a setup timing check.
>
> We see that the muxes result in two true paths, either first the top block and the bottom, or the bottom block at the top. We then take the maximum of these two paths:
>
> $$t_{period} = t_{clk,q} + t_{p,max} + t_{setup}$$
> $$= 10ps + 500ps + 300ps + 40ps = 850ps$$

Part b)

If this design runs at 1GHz, does it face any timing violations? If so, list them and explain.

> Yes, there are timing violations. This design has a hold time violation as
>
> $$t_{hold} > t_{clk,q} + t_{p,min} = 10ps + 20ps = 30ps$$

Part c)

You may now adjust the clock skew of the design. What is the clock skew that minimizes the clock minimum clock period while avoiding timing violations? What is the new minimum clock period? Show your work.

> We use the following formula to define the constraints we need to meet:
>
> $$t_{hold} + t_{skew} < t_{clk,q} + t_{p,min}$$
>
> In this case, we see that this holds when $t_{skew} < -70ps$, as
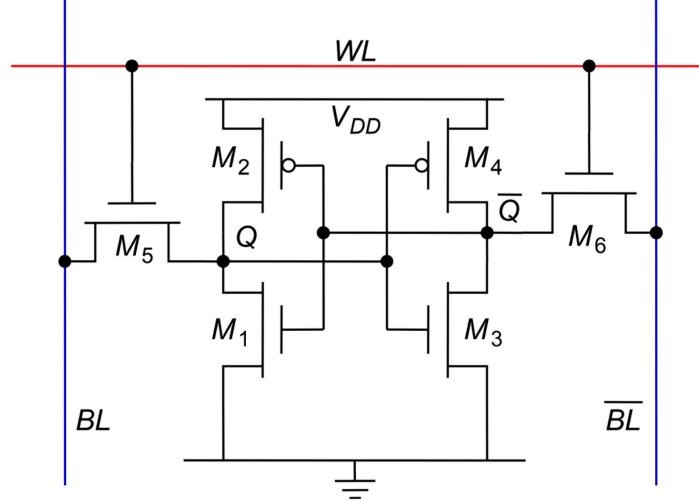>
> $$100ps - 70ps = 10ps + 20ps$$
>
> However, we need to recalculate the new clock period:
>
> $$t_{period} = t_{clk,q} + t_{p,max} + t_{setup} - t_{skew}$$
> $$= 10ps + 500ps + 300ps + 40ps - (-70ps) = 920ps$$

Problem 2: S(uper)RAM Design

Part a)

Consider the 6T SRAM design below. Assume that the SRAM operates at 1V, and that $V_{TH_N} = |V_{TH_P}| = 0.2V$.

i) You want to ensure that this SRAM can do non-destructive reads. Assuming you are reading a value of $Q = 1$, list any constraints on the relative sizes of transistors in this design.

> We read the SRAM by precharging the bit lines to 1. In this case, we detect which bitline is being discharged, which is on the side of the SRAM holding a value of 0, in this case $\bar{Q}$.
>
> We then see that we need to maintain that $\bar{Q} < V_{TH_N}$ so that the read is not destructive. In this case, $R_{M_6} > 4R_{M_3}$. This results in the following sizing constraint:
>
> $$4(W/L)_{M_6} < (W/L)_{M_3}$$

ii) You want to ensure that this SRAM is writable. If you are writing a value of $Q = 0$, list any constraints on the relative sizes of transistors in this design.

> In this case we want to make sure that the write path can overpower the feedback loop of the SRAM. In order to write a value of 0 we want to ensure that $Q < V_{TH_N}$ so we see that $4R_{M_5} < R_{M_2}$. This results in the following sizing constraint:
>
> $$(W/L)_{M_5} > 4(W/L)_{M_3}$$

Part b)

You are tasked with designing a single-ported 128-bit wide word-addressable (128 bit words) SRAM with a capacity of 32KiB (32×1024 Bytes). Consider the diagram below for reference.
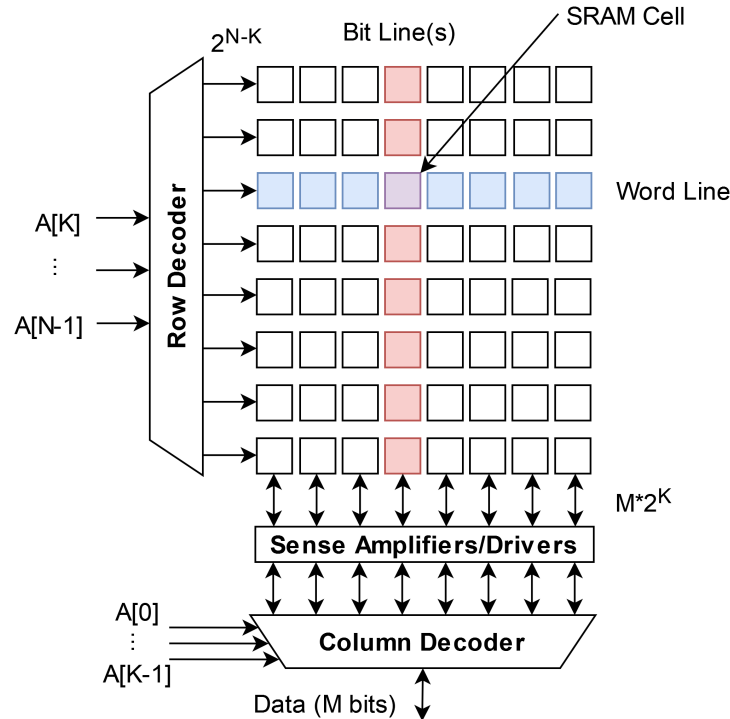
i) How many address bits are needed for this SRAM?

> To do this we find the number of words in the SRAM: $N = 32 \times 1024 \times 8/128 = 2048$
>
> The number of bits needed to address this is $N_{bits} = \log_2(N) = 11$

ii) Assume you want the SRAM cells to be arranged to have a 1:2 aspect ratio (that is, twice as many columns as rows). How many address bits are used for the row decoder? What about the column decoder? Show your work.

> Note: Due to an error when writing this question, the SRAM cannot be evenly divided into a 1:2 aspect ratio. An answer that mentions this is correct. Otherwise:
>
> With $32 \times 1024 \times 8 = 2^{18}$ SRAM cells, there will be $2^{8.5}$ columns and $2^{9.5}$ rows. With this, we need $\lceil 8.5 \rceil = 9$ bits for the row decoder, and $\lceil 9.5 \rceil - \log_2(128) = 3$ bits for the column decoder.

Problem 3: Love $$$

Part a)

You are given several options for implementing a 32KB cache, and decide to explore the effect of cache associativity on performance. Rank each of the following designs (ranking the best performing as 1st) for each of the metrics listed below. If equivalent, give the same ranking.

Cache Parameter	Direct Mapped	4-way set associative	Fully Associative
Hit rate			
Hit time			
Tag check hardware complexity			
Cache placement flexibility			
Cache replacement policy flexibility			

Cache Parameter	Direct Mapped	4-way set associative	Fully Associative
Hit rate	3	2	1
Hit time	1	2	3
Tag check hardware complexity	1	2	3
Cache placement flexibility	3	2	1
Cache replacement policy flexibility	3	2	1

Part b)

Calculate the number of bits in the offset, index, and tag fields for each of the following caches:

i)   32KB, direct mapped, 1KB block size, 32 bit words

> Offset: 8
> Index: 5
> tag: 17

ii)   1MB, 8-way set-associative, 4KB block size, 64 bit words

> Offset: 9
> Index: 5
> tag: 14

iii)   8KB, fully associative, 512B block size, 8 bit words

> Offset: 9
> Index: 0
> tag: 23