## Eksamen 2022

By Jesper Bertelsen

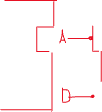
## Question 1

### Et billede, der indeholder Font/skrifttype, tekst, linje/række, typografi Automatisk genereret beskrivelseSketch a static CMOS gate computing the following equation. You may assume you have both true and complementary versions of the inputs available.

What to do:

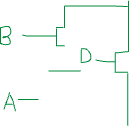
Things to know.

After that is done I will invert the signal.



Et billede, der indeholder Font/skrifttype, tekst, linje/række, typografi

Automatisk genereret beskrivelse



Normally you would then invert the signal, to get the non inverted signal, but as our function is the invertion of some function, we want the inverted signal, so no inverter is then added.

### 

Design the size of transistors in a way that tpdr=tpdf. Explain the different steps that you took to reach the final answer.

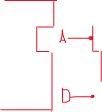
So my gameplan is to look for the worst case of the tpdf, find the width ratios so that the equivalent resistans in this path is R. From that I will make the width of the pmos transistors such that their equivalent resistans is R aswell. Knowing that pmos has less mobility than the nmos, I will make the ratio between them being twice as large of a pmos transistor width than the nmos transistor width.



The worst fall path is when



I look for when C is off, as a parallel



connection would fasten the delay.



For the nmos I set its width to be a factor

And



For *m* being a konstant I made to illustrate



Et billede, der indeholder Font/skrifttype, tekst, linje/række, typografi

Automatisk genereret beskrivelsethe mobility of the type of transistor.



For nmos this factor is 1.



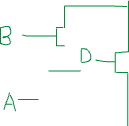
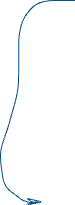
Indicating that the width of the nmos should



be 3\*w\_min

Now let’s look for the other alphas

For



So the alpha of C is the same as the alpha



of A.



Now for

The reason behind this path is because we don’t want the equivalent resistance to be more than 1. So by choosing the D = 1, other = 0 I ensure, that is

For the pull down network, the paths are almost the same.

The worst case can be either when , the last path just fastens the delay, as this path will become the second to make it to the output and then decrease the delay.

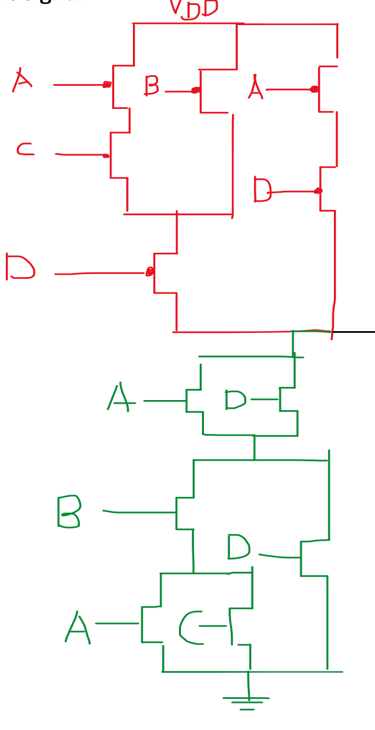
For the pmos we set the *m* factor to 2 to compensate for its lack in mobility.

We say that the width factors are the same:

So

This ensures, that has to be 4 aswell to fullfill the equation

And for the last path. C just adds to the contribution. I set it to 2 times the min width.

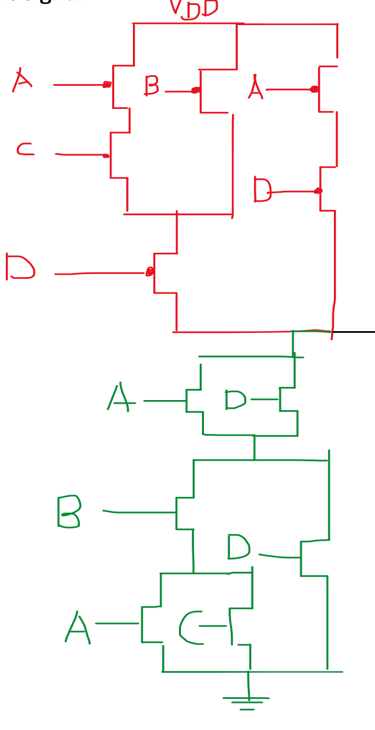
So it all put together:



The lengths are all the same = 60nm



### Make a RC model, then estimate

Let me make the RC equivalent:

For each transistor we replace it by a drain capacitance connected to a resistor connected to a source capacitance. The signal then acts as a switch



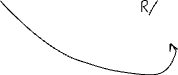
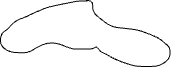
So let me do that.



As is designed to be equal to it doesn’t matter which signal change we want to check.

Let me check for when . The pulldown switches are closed and the pull up switches are open.

Simplifying the circuit for



The delay can be calculated with:

So for each node we will look at its capacitance, then times it with all the resistors it has seen.



Now substituting the values:

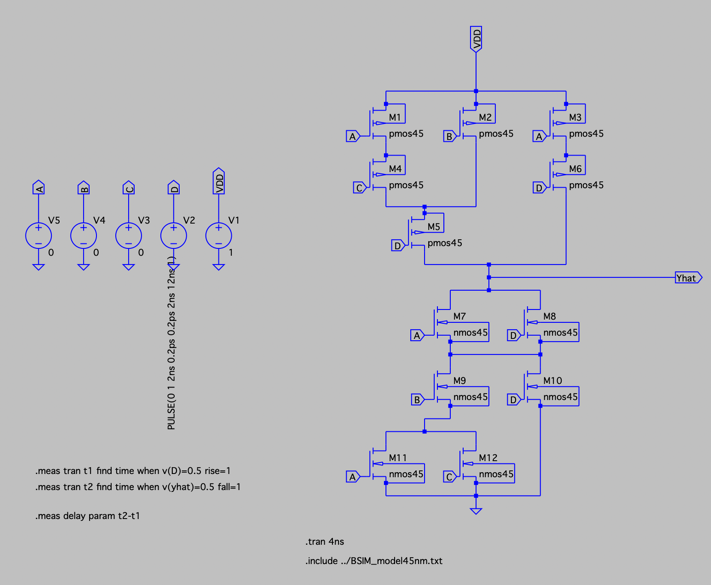


=============

=============

### Simulate the designed circuit in LTspice and check if tpdr=tpdf. In case tpdr≠tpdf, try to make them equal by changing the size of the transistors.

I made the setup in LTSPICE



I switch the , at 2ns, making a measurement t1 that checks when the rise of V(D) reached 0,5. A measurement t2 then checks the time when our function V(yhat) falls and reaches the value 0,5V.

Delay then tells the difference in time.

The pulse works in LTSPICE works a little weird. It needs a rise/fall delay. When it doesn’t, it simply just takes the on value and divides it by 10, and that value is the rise fall delay of the input.

Et billede, der indeholder tekst, Font/skrifttype, skærmbillede, typografi

Automatisk genereret beskrivelseI wanted to make the rise fall delay as small as possible for the input, as it’s not the inputs delay we want to measure.

Figure 1: Delay of input rise and fall delay set to 2ps

To achieve an estimate of this, I set the inputs rise and fall delay to 2ps and then 0,2ps. Both times reaching a of around 6ps, so this must be the delay of the circuit.

Compared to my elmore delay model, this seems a lot quicker than I would have otherwise expected.

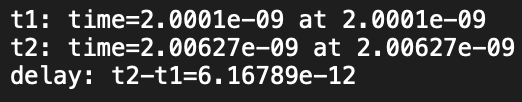
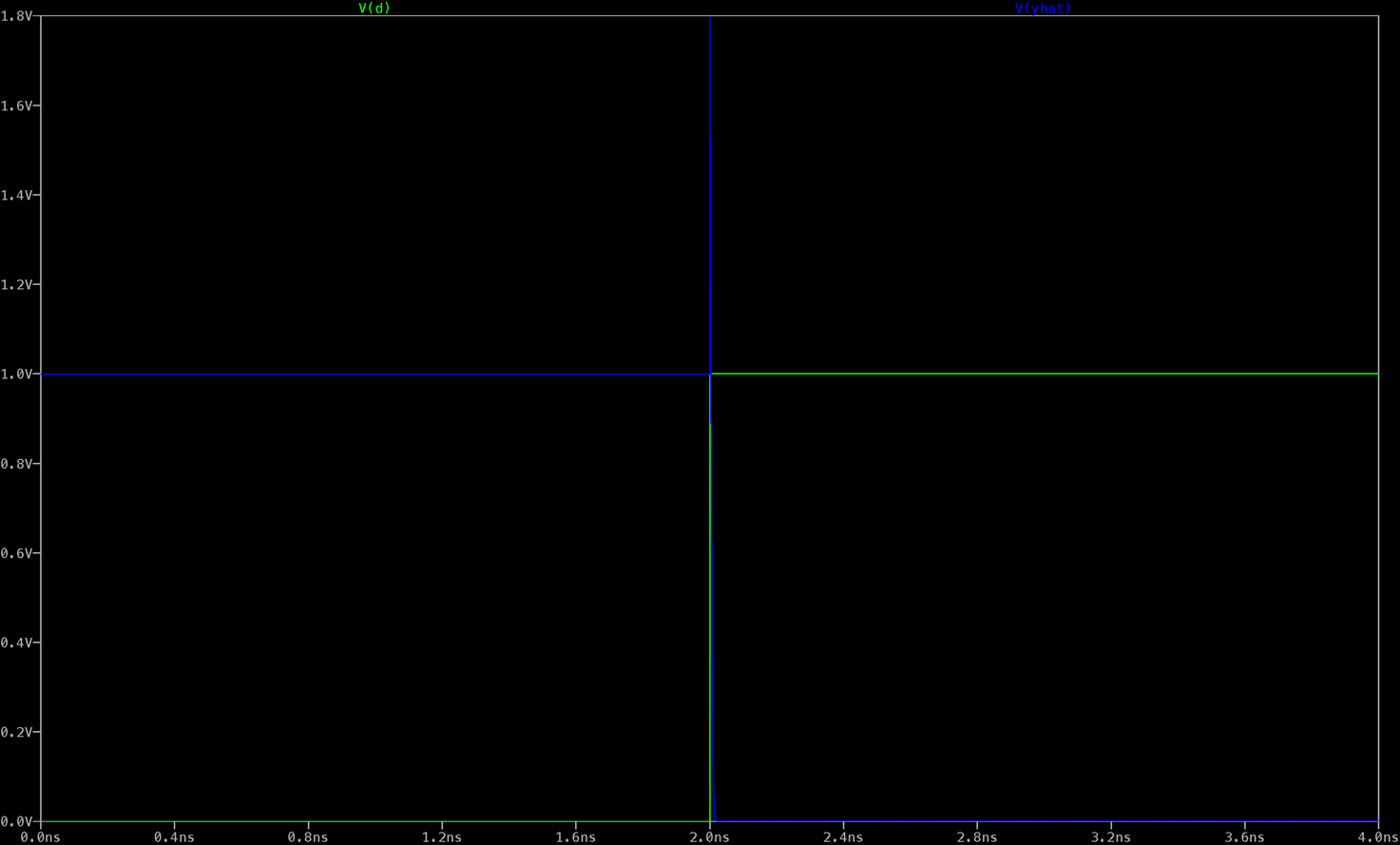


Figure 2: Delay of input rise and fall delay set to 0,2ps

The measurement telling us, that the delay is as fast than I would have otherwise expected.

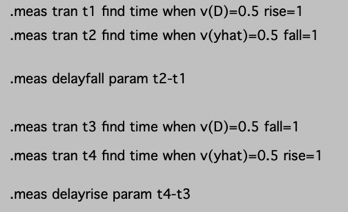
The simulation.

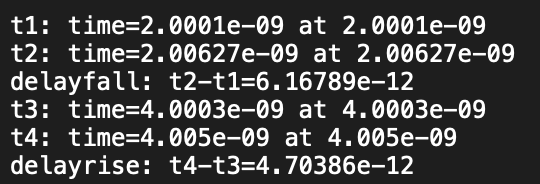


It’s hard to see the delay from High to low. As the input value is set to rise very quickly, the output voltage spikes, which one can see.

Now adding more measurement to see the difference between rise and fall delay.

The measurements:





There is a small difference in delays.

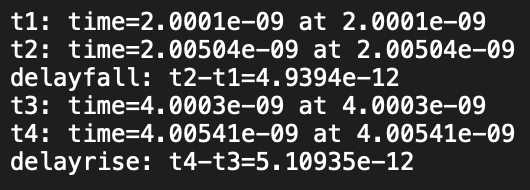
But the question is, if this really matters a lot.

The difference is

Making the rise delay

Quicker than the fall delay.   
This circuit is not a huge complex one, so this might be neglectable, but for larger circuits, this could mean a lot.   
  
I will try to adjust some sizes of the transistors to see if I can achieve a better delay:

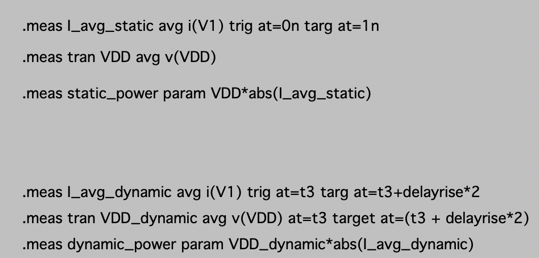
Adjusting the D\_nmos from 2\*w\_min -> 3 \* w\_min resulted in:



Which is almost the same. But this could have a negative effect on other combinations, so this might not be desired.

### Evaluate the static and dynamic power consumptions for your circuit in LTspice. For dynamic power consumption, change the input from ABCD=”1001” to ABCD=”0100”.

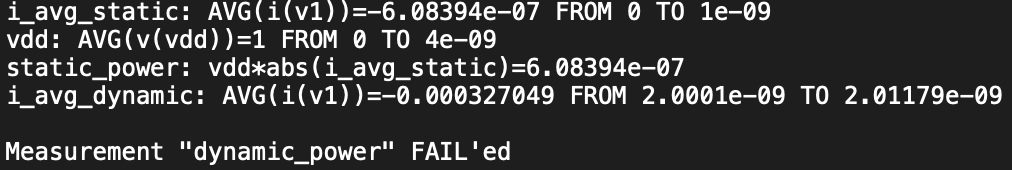
Making new measurements.   
For the static power consumption, and for the dynamic power consumption.



The static is straight forward. Measure current at an interval where the circuit isn’t changing. Then take the average VDD to find the static power at .

For the dynamic I wanted to find the current average doing the switch. I did this with the period [t3; t3 + delayrise\*2] assuming that the rise would be symmetric and 2 \* delayrise resulting in going from 0V -> 1V.

I tried to do the same with the voltage, but somehow the measurement wouldn’t add it self, making the dynamic power fail, as there were no voltage paramater.



Making a very rough estimate, saying that the voltage during this time was linear:

As said, this is a very rough estimate, but I didn’t think it would have been so bad. This estimate is nowhere need anything I can use.

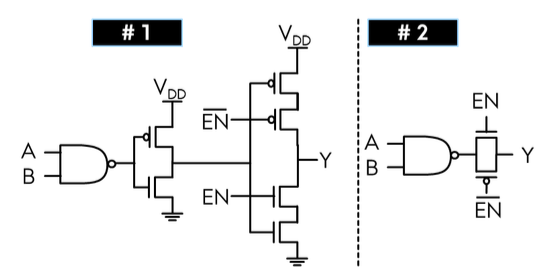
Results  
==================

Static power = 0,6mW

==================

Husk at aflevere filen også!

## Question 2





### What is the static gate that the following CMOS circuits implement?

The first implementation:

The second implementation

### Describe the advantage(s) and disadvantage(s) of configuration #1 compared with #2.

If the output of the nand gates are the output y, then the

1. implementation = , when en = 1, the circuit is shorted otherwise.
2. implementation = , when en = 1, the circuit is shorted otherwise.

So what the implementation wants is to have a switch. The switch with cmos technology inverts the signal.

The first implementation doesn’t want it to be inverted, so it puts in an inverter to make it a switch acting like a buffer.

The second implementation doesn’t mind the inversion, so it just keeps the inverted signal.

## Question 3

### Write a VHDL code for implementing the following truth table that has three inputs (i2, i1, and i0) and two outputs (a1, a0).

Et billede, der indeholder skærmbillede, tekst, Font/skrifttype, nummer/tal

Automatisk genereret beskrivelse



I could find the function using a K map and make the architecture dataflow, but a behavioral code can do it too, and this might be more intuitive for someone. HOWEVER what I notice is, that this is just a 4-2bit encoder.

The code:

-----------------------------------------------------------------------------------------------------------------------------------

;

-----------------------------------------------------------------------------------------------------------------------------------

The process might be needed for the combinational logic, I don’t fully understand when to and when not to process yet.

Without it I cannot make a bitstream and with it I can, so I guess I need to.

### Then, write a test bench for it in order to verify the operation of your VHDL code.

-----------------------------------------------------------------------------------------------------------------------------------

# Making a process for the entity I want to test:

# Making a process for the input combinations:

# And then just repeating for every expected combination.

-----------------------------------------------------------------------------------------------------------------------------------

This is checking for the cases we specified. Actually there are 16 combinations, that we should check for if we were to do it right.