Written examination for

MCC092 and MCC091 Introduction to Integrated Circuit Design

Monday August 27, 2018, at 8.30-13.30 at the M building

Staff on duty: Lena Peterson, CSE dept, phone ext: 1822, or mobile 0706-268907. Lena will visit around 9.30 and 12.00.

Administration: Send exams to Lena Peterson CSE dept. Send lists to CSE student administration office.

Allowed technical aids for students: This is a closed-book exam. Allowed aids: A Chalmers-allowed calculator (non graph-drawing) plus pencil, eraser, ruler, and dictionary (these are always allowed).

Results: The results from the examination will be sent to you via the Ladok system within three weeks. The grading review will take place Monday September 17, 12.30-13.15 in room 4128 at the CSE department.

Solutions: Solutions will be posted on the course web site in PingPong no later than Tuesday August 28. Any student who does not have access to the 2017 course web site can contact Lena Peterson (via e-mail to lenap@chalmers.se) to obtain the solutions.

Instructions:

- Write legibly.
- State any assumptions you make.
- Explain your reasoning and calculations (except when the problem says otherwise). Partial credits can be awarded, but if we do not understand you that is not possible.
- Number all pages and write your code on each page.
- Put only one problem per page.
- Do not write in red since that is the color used for the grading.

Good luck!

Grades:

The written examination contains six problems, each worth 10 points. You need 30 points to pass (grade "3"), at least 40 points for grade "4" and at least 50 points for grade "5". Bonus points from the fall 2017 course instance will be added before the higher grades are assigned.

Problem 1: Five short questions about power and wires

a) In modern CMOS processes the static power consumption has become an increasing problem. Why is it a much larger problem now, than, say, fifteen years ago? Give at least two reasons. (2 p)

b) Write down the expression for the switching power per transistor gate in CMOS. What scaling is expected of this power when a CMOS fabrication process is scaled according to the Dennard scaling rules, with a scaling factor S? (Where S is usually assumed to be $\sqrt{2}$ per CMOS process generation). Motivate your reply by giving the scaling, expressed in S, of each of the factors in the expression for the switching power. (3 p)

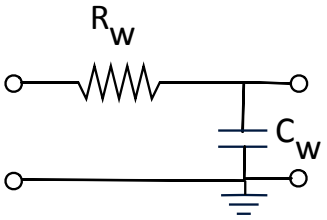


Figure 1: A suggested wire model.

c) Why is the suggested model shown in Figure 1 for an on-chip wire segment with total resistance R_w and total capacitance C_w not useful for delay calculations? (1 p)

d) What model do we usually use instead for a wire segment? Why is it preferable? (1 p)

e) Draw the circuit model for an inverter, with resistance R, input capacitance C and parasitic delay p_{inv} driving an identical inverter across a wire segement, with resistance R_w and capacitance C_w. From your diagram derive the expression for the delay from input of the first inverter to the input of the second inverter. (3 p)

Problem 2: Logical functions and layout

A 2-input exclusive or (XOR) gate can be implemented using a 2-input NOR gate and a 2-input AND-OR-INVERT gate connected as shown in Figure 2.

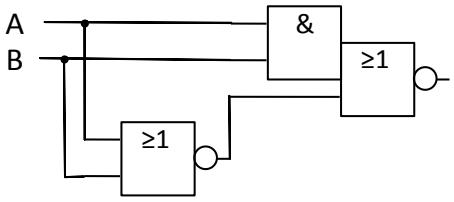


Figure 2: An XOR gate made up from two logic gates.

a) Draw the transistor diagram for the 2-input XOR gate consisting of the two gates shown in Figure 2. (2 p)

b) Draw the transistor diagram of the direct implementation of a 2-input XOR gate derived from a truth table or Karnaugh digram. Identify at least one advantage of the solution in Figure 2 over this solution. (2 p)

c) Draw the layout of the transistor diagram you drew in task a) in the supplied template. You are only allowed one connection point from the outside to each of the input signals A and B. Use the template supplied in Figure 3. It is repeated again at the end of this exam on tear-off pages. (6 p)

d) BONUS Draw the logic diagram of the exclusive negative or (XNOR) gate, consisting of two logic gates, corresponding to the XOR gate shown in Figure 2. Verify that your solution has the correct logic function. (3 p)

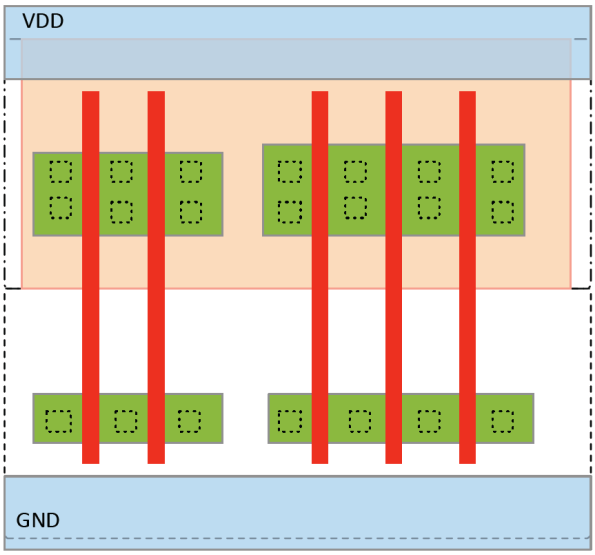


Figure 3: A template for the layout of the XOR gate.

Problem 3: Logical effort, parasitic effort

Again consider the transistor diagram that you drew in problem 2 a). Assume that the transistors are scaled so that the worst-case resistances are the same for all paths in both gates. For simplicity we call this resistance R.

a) Draw the transistor diagram again with all transistor widths showing the correct scaling. Assume that the narrowest transistor has width 1. (2 p)

b) What are the parasitic delays of the NOR gate and the AND-OR-INVERT gate with these transistor widths? (2 p)

c) Your friend Ernie proposes that the logical effort for input A should be calculated as the sum of the input capacitances for node A in the NOR gate plus the input capacitances for node A in the AND-OR-INVERT gate divided by the input capacitance of an inverter with resistance R. Argue for why this cannot be correct. (2 p)

d) Consider the slightly different circuit in Figure 4 with four inputs. Calculate the logical effort for input A, g_A and the corresponding parasitic delay p, for this circuit. (4 p)

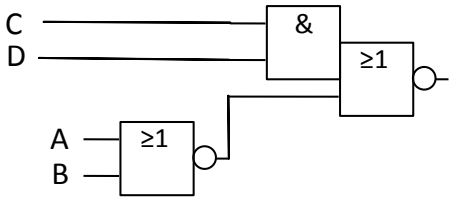


Figure 4: Another circuit made up from the same two logic gates.

Problem 4: Clock distribution, path delay

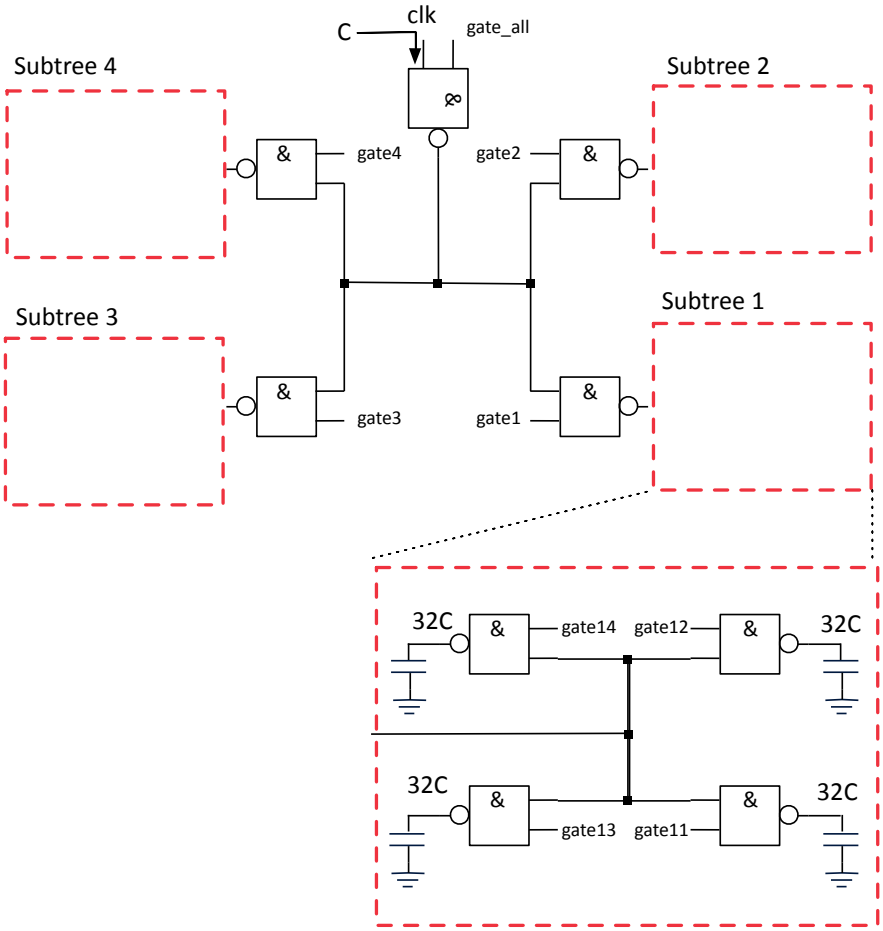


Figure 5: Gated clock buffer tree. The NAND2 gate at the root of the tree has the input capacitance C. The four subtrees are identical, but for simplicity we have only drawn one of them.

a) A gated clock buffer tree is shown in Figure 5. Its purpose is distribute the clock signal to 16 similar circuit blocks on a chip, while also facilitating clock gating for each of the 16 blocks (or for larger sections of the chip). Your task is to determine the sizes (input capacitances) of the 2-input NAND gates that are used to gate the clock signal in the tree for minimum delay. Also calculate the resulting minimum delay. (5 p)

b) You propose to improve the buffer tree design by adding a fourth stage to the buffer. This stage does not have to be gated, so it can be an inverter stage (the clock being inverted is not a problem, since it will be inverted everywhere). This inverter stage could be added at the beginning or the end of the buffer tree. Choose one of these two designs and motivate your choice! Calculate the resulting minimum delay with optimal sizing. Assess your proposal - was it a good idea to add one more stage? (5 p)

Problem 5: Sequencing **Note that students taking this exam for the old course, MCC091, can choose to solve problem 7 instead of problem 5.**

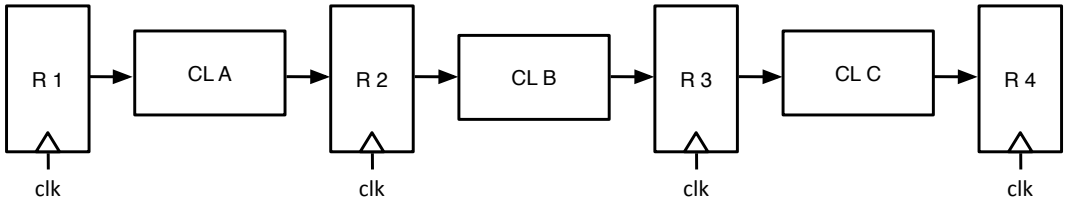


Figure 6: Pipeline comprising four registers, R 1 through 4, and three combinational-logic block, CL A through C.

Consider the pipelined logic shown in Figure 6, where the inputs and outputs of the three combinational-logic blocks A, B and C are sequenced by four registers numbered 1-4. The timing data for the logic blocks are given in Table 1. The four registers are identical. The timing data for the flip-plos in the registers are given in Table 2. The clock frequency is 2 GHz.

Table 1: Timing parameters for the combinational logic blocks

Timing parameter	Block A [ps]	Block B [ps]	Block C [ps]
Propagation delay, t_{pd}	400	200	50
Contamination delay, t_{cd}	10	100	30

Table 2: Flip-flop timing characteristics

Flip-flop timing parameter	Value [ps]
Setup time, t_{setup}	-30
clk-to-Q propagation delay, t_{pcq}	80
clk-to-Q contamination delay, t_{ccq}	40
Hold time, t_{hold}	10

a) Determine if there will be a setup violation in the pipeline with the given timing data. (3 p)

b) Determine if there will be a hold violation in the pipeline with the given timing data. (3 p)

c) Discuss what you as a designer can do to remove a setup or hold violation, in general. If there is any setup or hold violation in the given pipeline also determine the changes that need to be implemented in this particular case, to remove these violations. When possible give numbers for the changes that are required. (4 p)

Problem 6: Prefix adders

In Figure 7 you see a figure of a prefix adder of an unknown type. The part that is drawn in detail is only the PG tree structure.

a) Write down the logical expression for sum bit i, expressed in block propagate and block generate signals from any prefix-adder tree. (2 p)

b) If we assume just straight lines in the white space at the bottom of the tree in Figure 7, which of the 32 sum bits cannot be formed from the tree as shown in Figure 7? (2 p)

c) Complete the adder schematics supplied below with grey cells so that all sum bits can be formed. Draw your solution in the attached schematic which you should tear of and hand in with your solution. The schematic is repeated on tear-off pages at the end of the exam for your convenience. Any schematic that creates all the necessary signals will be considered correct. (3 p)

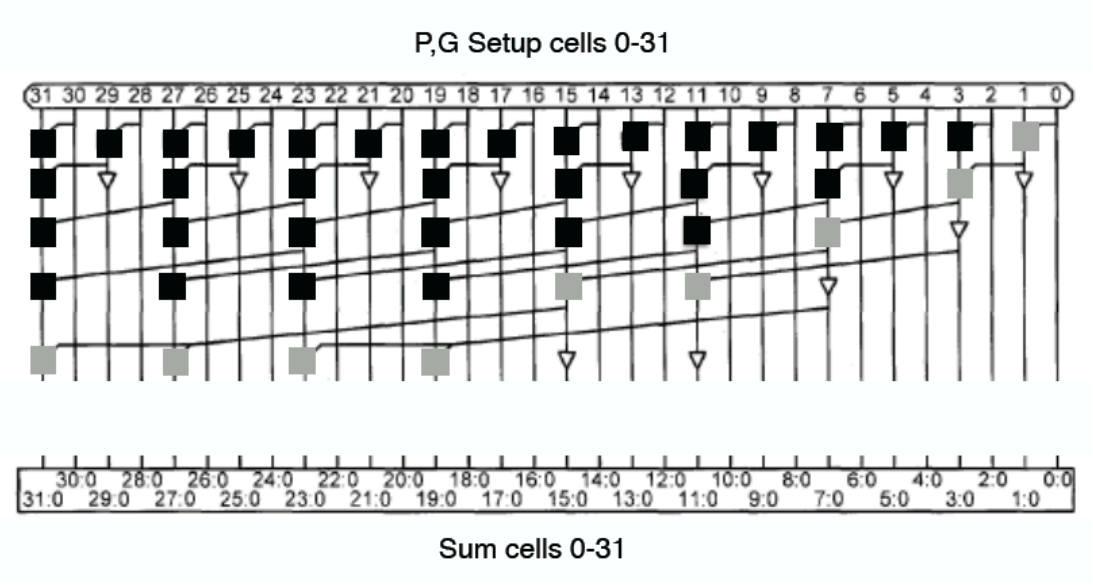


Figure 7: A 32-bit prefix adder of an unknown type where part of the tree is missing.

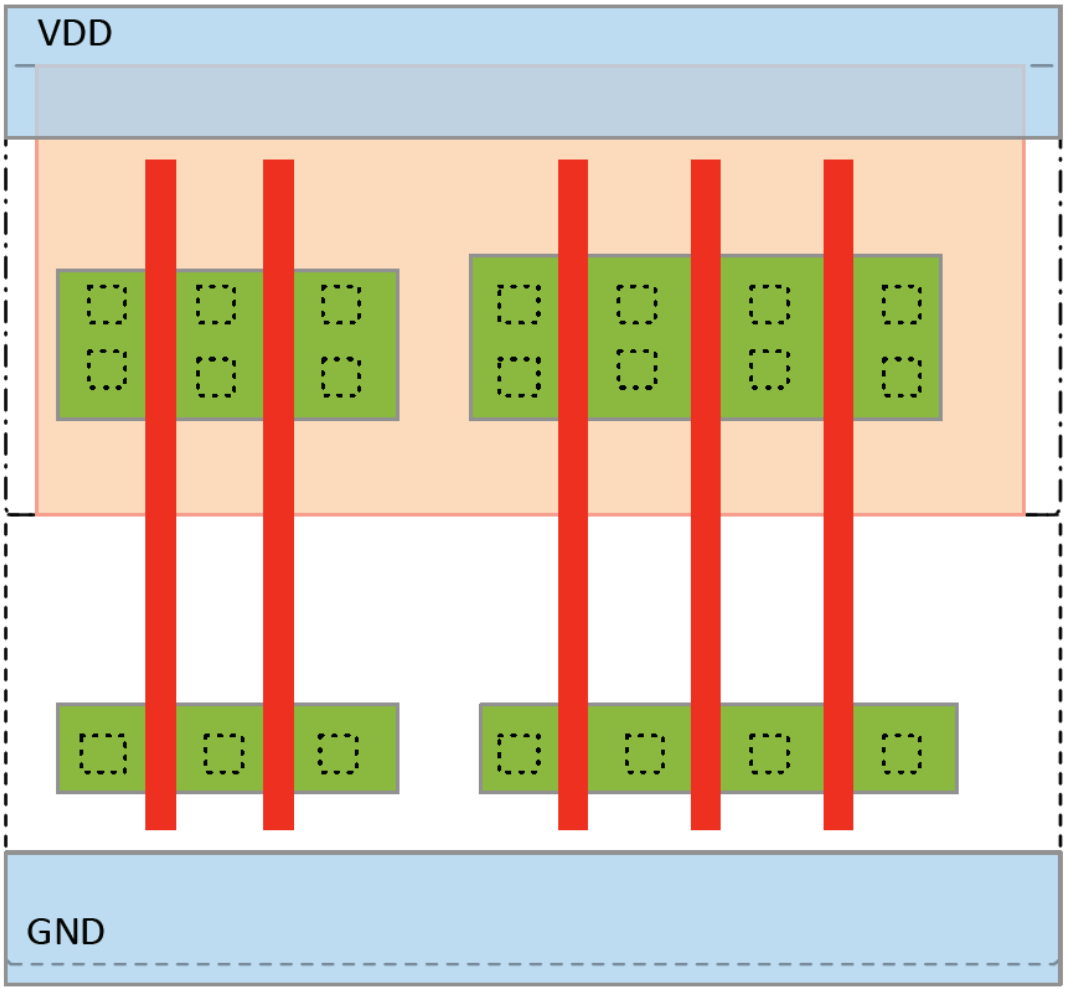
d) Assume (not quite correctly) that all black and grey PG cells have the same propagation delay, called 1 unit delay. Which of the 32 sum bits will then be last ones to be done in your solution from task c)? List ALL of the sum bits that have this the longest propagation delay. What is this the longest propagation delay through the prefix-adder tree expressed in unit delays? Ignore the buffers (triangles) shown Figure 7 when you find these critical paths. (3 p)

Problem 7: Switching voltages, activity factor **NOTE! This problem is only for students who are taking this exam for the old course MCC091 and who opt not to do problem 5.**

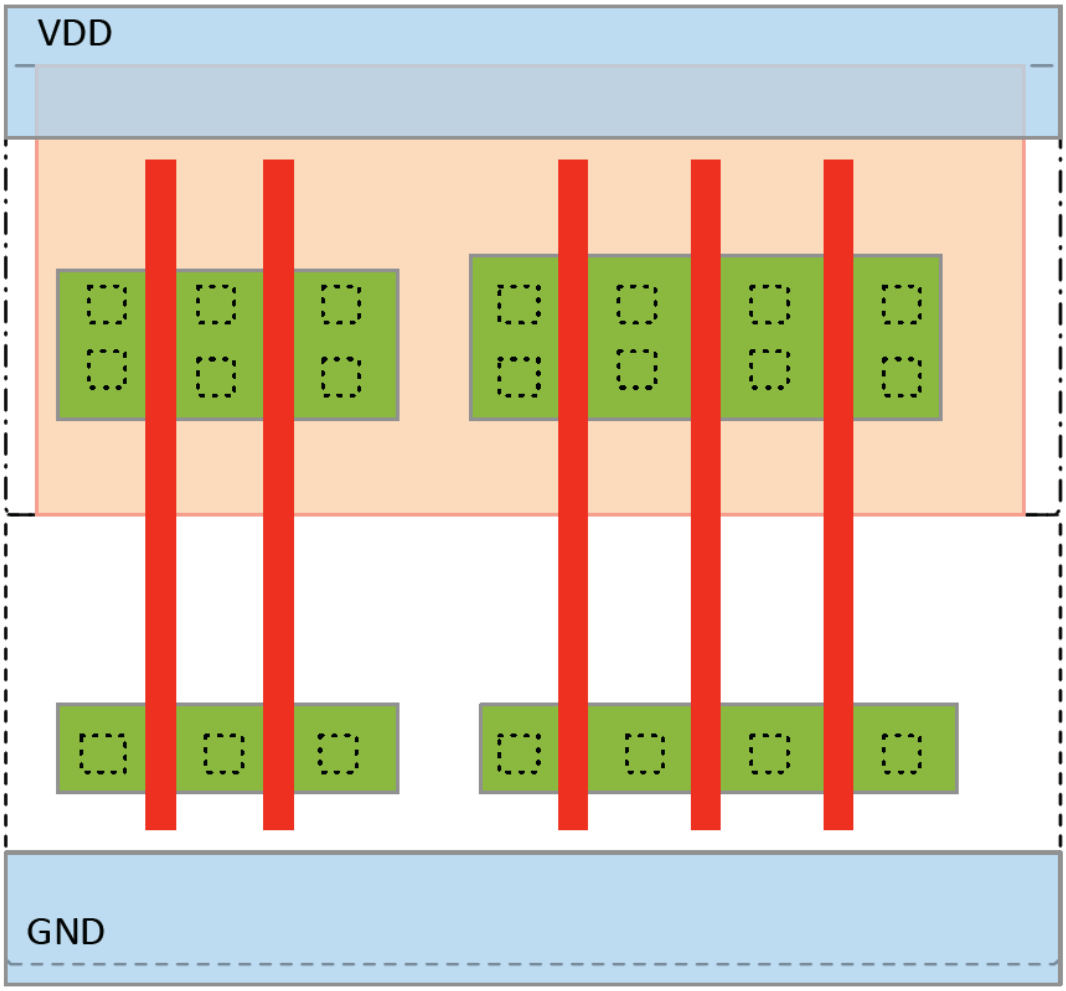
a) What if we connect the three inputs of a 3-input NAND gate, which has its n-net and p-net properly scaled for the same worst-case resistance, to use it as an inverter. If we sweep the input voltage to obtain the voltage transfer curve, what switching voltage, V_{sw}, would we expect to find, expressed in V_{DD}, V_{TN} and V_{TP}? (7 p)

b) Consider the AND-OR-INVERT 2+1 gate. What is the expression for the probability that the output of this gate is "one"? What is the resulting probability value if we assume $P_A = P_B = P_C = 0.5$ for the three inputs? What is then the resulting activity factor for the gate? (3 p)

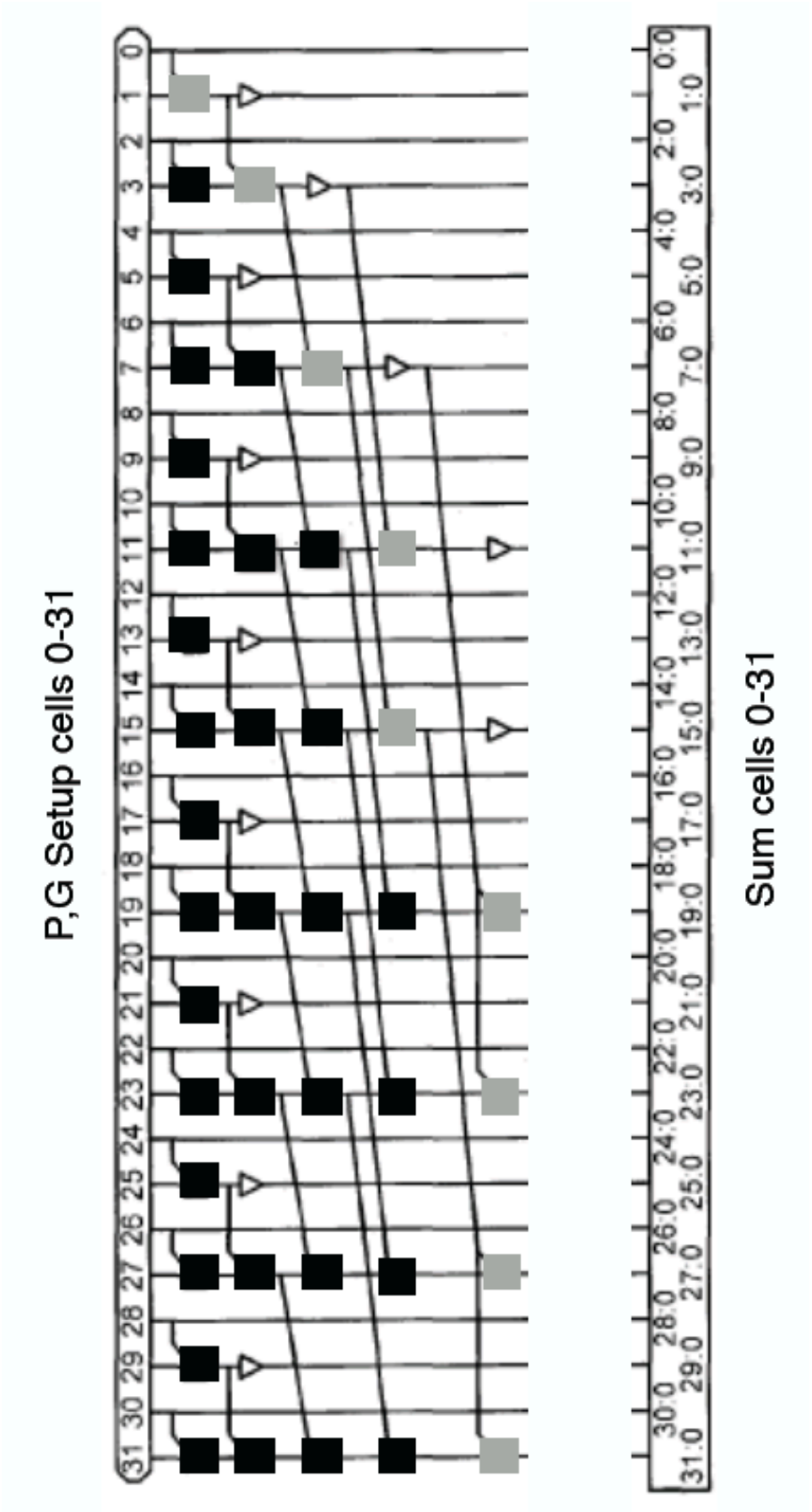
MCC092 2018-08-27 Tear-off page for task 2 c). Write your anonymous code here:



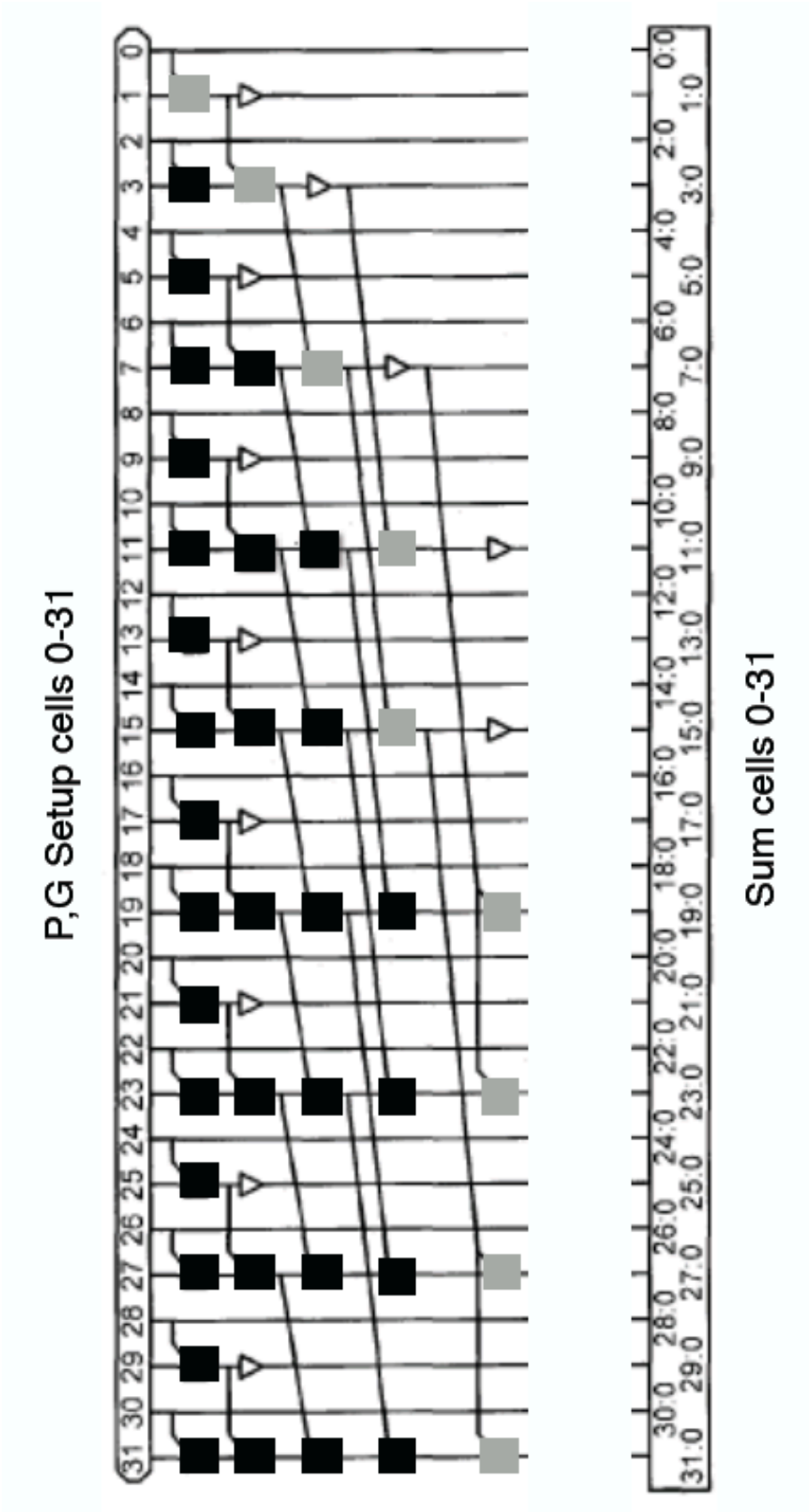
MCC092 2018-08-27 Tear-off page for task 2 c). Write your anonymous code here:



MCC092 2018-08-27 Tear-off page for task 6 c). Write your anonymous code here:



MCC092 2018-08-27 Tear-off page for task 6 c). Write your anonymous code here:



Solution for written examination for

MCC092 Introduction to Integrated Circuit Design, August 27, 2018

Solution 1: Power and wires

a) Reason 1: Due to the scaling down of the supply voltage the transistor threshold voltages also have to be scaled down to maintain circuit speed. The scaling down of the threshold voltages makes the ratio of the off current to the on current smaller and thus the leakage power larger in relation to the switching power. Reason 2: Chips are getting larger and larger with more of the circuitry being unused at particular point in time. The unused circuitry contributes mainly leakage power (at least if clock gating is used for unused circuitry).

b) The equation for the switching power is

$$P_{sw} = fCV_{\mathrm{DD}}^2 \tag{1}$$

Dennard scaling with a factor S scales the switching power with a factor $1/S^2$. We can derive this from the scaling of the three factors in the expression for the switching power. The clock frequency f scales as S (because it is inversely proportional to the gate delay τ which scales as $1/S$), The gate capacitance, C, scales as $1/S$ and the supply voltage also scales as $1/S$. From these three factors we find that the scaling is $1/S^2$.

c) In an on-chip wire the capacitance and resistance is distributed along this wire. This model is not good because it overestimates the delay when it (incorrectly) places all the capacitance at the far end of the wire.

d) We usually use the pi model which places half the wire capacitance at the near end and half of it at the far end of the wire resistance, thus modelling the distributed nature more correctly with lumped components.

e) A figure will be added here later.

The resulting propagation delay using the Elmore delay model is:

$$t_{pd} = 0.7\left[R \times (C + C_w + C) + R_w \times (\frac{C_w}{2} + C)\right] = 0.7\left[(R + \frac{R_w}{2}) \times (2C + C_w)\right] \tag{2}$$

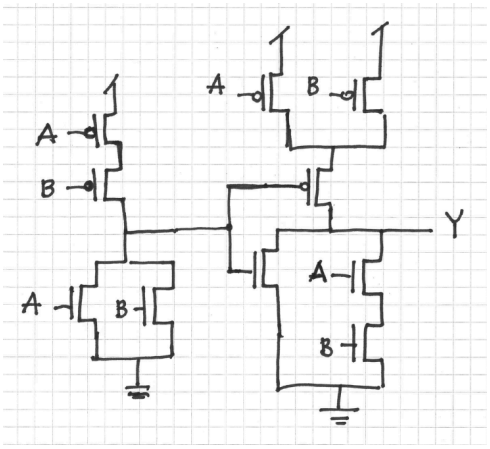


Figure 8: The transistor diagram for the XOR gate in problem 2 a).

Solution 2: Logical functions and layout

a) The circuit diagram is shown in Figure 8.

b) The circuit diagram can be seen in Figure 9. The most obvious advantage of the solution in a) is that is as fewer transistors and no need for the inverses of A and B.

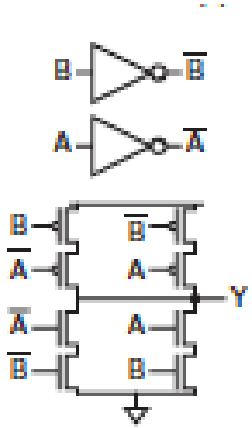


Figure 9: The straight-forward implementation of the 2-input XOR gate.

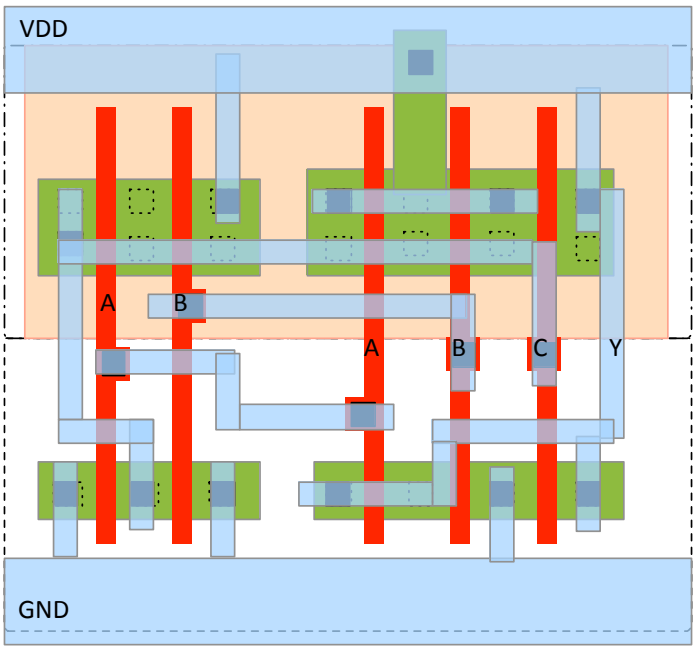


Figure 10: One possible solution for the layout of the XOR gate.

c) BONUS. To be added.

Solution 3: Logical effort, parasitic effort

a) The solution is shown in Figure 12.

b) This solution is also shown in Figure 12.

c) It cannot be correct because there are two different paths from inputs A and B to the output Y. Even though all resistances are the same the value you would get for the logical effort, g, would not give the critical delay from the equation $d = g \times h + p$. In cases where there is multiple paths from a particular input to the output this simple model does not work well.

d) The normalized delay for the path through the two gates is given by

$$d = p_{\mathrm{NOR}} + g_{\mathrm{NOR}} h_{\mathrm{NOR}} + p_{\mathrm{AOI}} + g_{\mathrm{AOIc}} h_{\mathrm{AOI}}$$

where only h_{AOI} depends on the load of the combined gate. From task a) we have $C_{\mathrm{INNOR}} = 5C$ and $C_{\mathrm{INAOIc}} = 5C$ so we do not need to do any rescaling due to a difference in input capacitance in the two gates.

We find

$$g_{\text{NOR}} = \frac{(4+1)C}{3C} = \frac{5}{3}$$

and

$$g_{\text{AOIc}} = \frac{(4+1)C}{3C} = \frac{5}{3}$$

Thus, we have

$$d = 2p_{\text{inv}} + \frac{5}{3} \times \frac{5C}{5C} + \frac{7}{3}p_{\text{inv}} + \frac{5}{3} \times \frac{C_L}{5C}$$

Thus find find that a model for the entire path has:

$$p_{\text{path}} = \frac{13}{3}p_{\text{inv}} + \frac{5}{3}$$

and

$$g_{\text{path}} = \frac{5}{3}$$

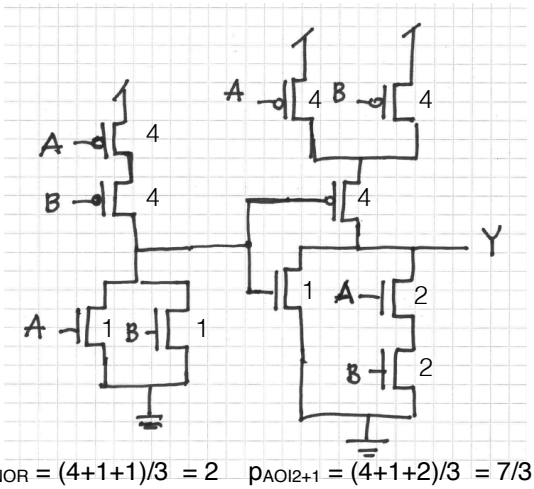


Figure 11: The transistor diagram for the XOR gate in problem 2 a).

Solution 4: Clock distribution, path delay

a) This problem can most easily be solved as a path delay problem. The path effort is:

$$F = G \times B \times H$$

In this problem we find:

$$G = \frac{4}{3} \times \frac{4}{3} \times \frac{4}{3}$$
$$B = 4 \times 4$$
$$H = \frac{32C}{C}$$

Thus, we have

$$F = \frac{4}{3} \times \frac{4}{3} \times \frac{4}{3} \times 4 \times 4 \times 32$$
$$F = \left(\frac{32}{3}\right)^3$$

The path has three stages so by inspection we find:

$$f_{opt} = \sqrt[3]{F} = \frac{32}{3}$$

The resulting path delay with three stages is:

$$D_3 = 3\left(\frac{32}{3} + 2p_{\text{inv}}\right) = 32 + 6p_{\text{inv}}$$

The resulting gate sizes can be found either by starting from the beginning or the end of the path. Starting from the end we find for stage 3:

$$\frac{32}{3} = \frac{4}{3} \times \frac{32C}{C_{IN3}}$$
$$C_{IN3} = \frac{4}{3} \times \frac{3}{32} \times 32C$$
$$C_{IN3} = 4C$$

And for stage 2:

$$\frac{32}{3} = \frac{4}{3} \times \frac{4 \times 4C}{C_{IN2}}$$
$$C_{IN2} = \frac{4}{3} \times \frac{3}{32} \times 16C$$
$$C_{IN2} = 2C$$

The solution is also shown in figure**??**.

b) There are good arguments both ways. If we place the inverter first we only need one inverter; on the other hand we have to size up all the NAND2 gates so the total area used is most likely larger than for the other solution. If we place inverters at the end of the path we need 16 inverters, but we make the more complex gates smaller since they come first. Both are valid arguments, but we should really investigate what the area difference is to make the decision. Both solutions will result in the same path delay.

We add one stage to the path with $g = 1$. So the path effort stays the same but we have four stages rather than three:

$$f_{\text{opt4}} = \sqrt[4]{\left(\frac{32}{3}\right)^3} = 5.9$$

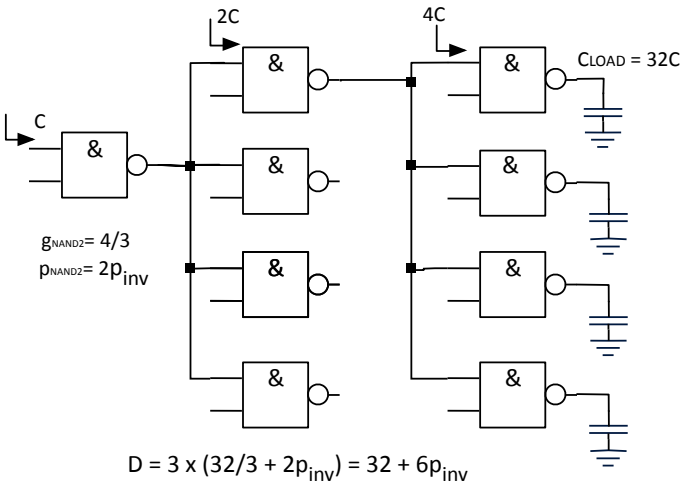


Figure 12: The clock-tree path delay problem with final sizes and the path delay computed.

The path delay with four stages is:

$$D_4 = 4 \times 5.9 + (3 \times 2 + 1)p_{\text{inv}} = 23.6 + 7p_{\text{inv}}$$

So the gain from adding one more stage was significant. If we assume $p_{\text{inv}} = 1$ the decrease in path delay is around 20 %.

Add an area comparison.

Solution 5: Sequencing

The clock frequency is 2 GHz which corresponds to a clock period $T_c = 500$ ps.

a) The condition that has to be fulfilled for all logic blocks in the pipeline to avoid a setup violation is:

$$T_c \geq t_{pcq} + t_{pd} + t_{\text{setup}} \tag{3}$$

We can also express that as

$$T_c \geq t_{pcq} + t_{pd\text{max}} + t_{\text{setup}} \tag{4}$$

For blocks A, B and C we have

$$t_{pd\text{max}} = 400\,\text{ps}. \tag{5}$$

With timing data for the present register we get:

$$500\,\text{ps} \geq 80\,\text{ps} + 400\,\text{ps} + -30\,\text{ps} \tag{6}$$

which we can simplify to

$$500\,\text{ps} \geq 450\,\text{ps} \tag{7}$$

Thus, we can conclude that with a clock frequency of 2 GHz we will not get any setup violation (unless there is clock skew somewhere in the system).

b) To avoid a hold violation the condition that has to be fulfilled for all logic blocks in the pipeline is:

$$t_{ccq} + t_{cd} \geq t_{\text{hold}} \tag{8}$$

We can also express that as

$$t_{ccq} + t_{cd\text{min}} \geq t_{\text{hold}} \tag{9}$$

For blocks A, B and C we have

$$t_{cd\text{min}} = 10\,\text{ps}. \tag{10}$$

With timing data for the present register we get:

$$40\,\text{ps} + 10\,\text{ps} \geq 10\,\text{ps} \tag{11}$$

There is no timing violation here either.

(I had intended there to be a violation, but I copied the data into the table incorrectly! I intended t_{hold} to be 80 ps. In that case we would get

$$40\,\text{ps} + 10\,\text{ps} \geq 80\,\text{ps} \tag{12}$$

which is not true.).

c) A setup violation is when a token does not reach its destination (the flip-flop) on time. A setup violation can be remedied AFTER implementation by increasing the clock period, that is decreasing the clock frequency. That way the token has longer times to reach its destination. At design time, the designer can redesign the offending logic blocks (the ones with the longest propagation delay) to decrease their propagation delay.

A hold violation is when two tokens merge in the pipeline. Such a violation cannot be remedied AFTER implementation; that is it does not help to increase the clock period. the At design time the designer has to increase the contamination delay, that is the shortest delay, for those logic blocks that are too fast.

Solution 6: Prefix adders

a) The sum is expressed in propagate and generate signals as:

$$S_i = P_i \oplus G_{i-1:0} \tag{13}$$

This expression can also be written as:

$$S_i = A_i \oplus B_i \oplus G_{i-1:0} \tag{14}$$

b) The missing outputs are number 30, 29, 28, 26, 25, 24, 22, 21, 20, 18, 17, 16, 14, 13, 12, 10, 9, 8, 6, 5, 4 and 2. Consequently, the sums that cannot be formed are the ones one step higher so 31, 30 , 29 and so on.

c) See solution in Figure 13.

d) The longest path in the complete adder in 13 has seven PG cells, that is the delay through the tree is 7 unit delays. It is three outputs that have this worst-case delay: 30, 26 and 22. For one output there can be multiple paths of the same length. In Fig 14 one path for each of these three outputs is colored. With a different solution for task c) the solution to task d) may be different.

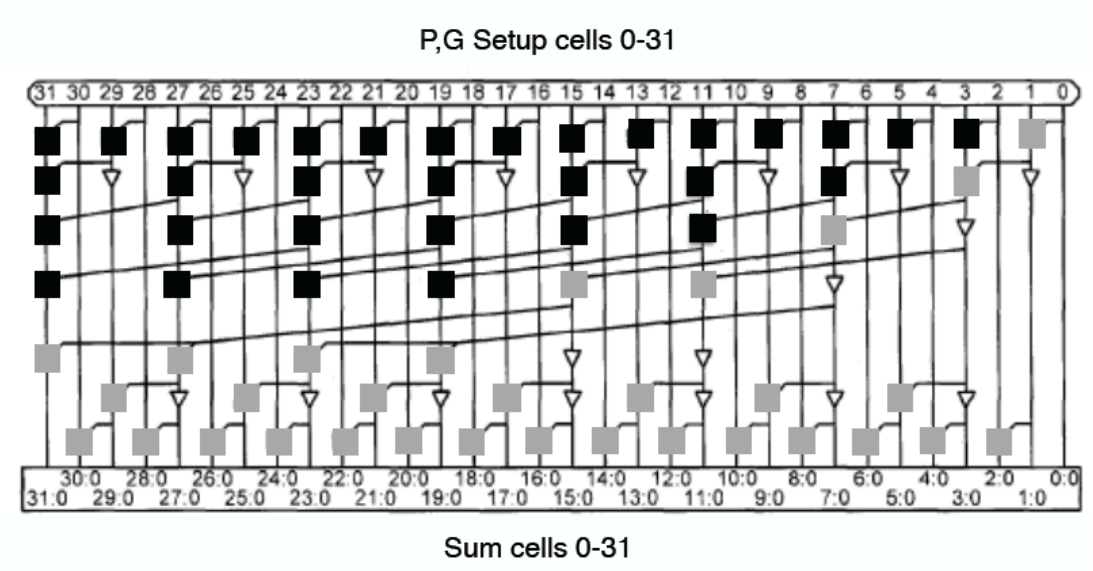


Figure 13: The complete 32-bit adder. Other solutions are also possible.

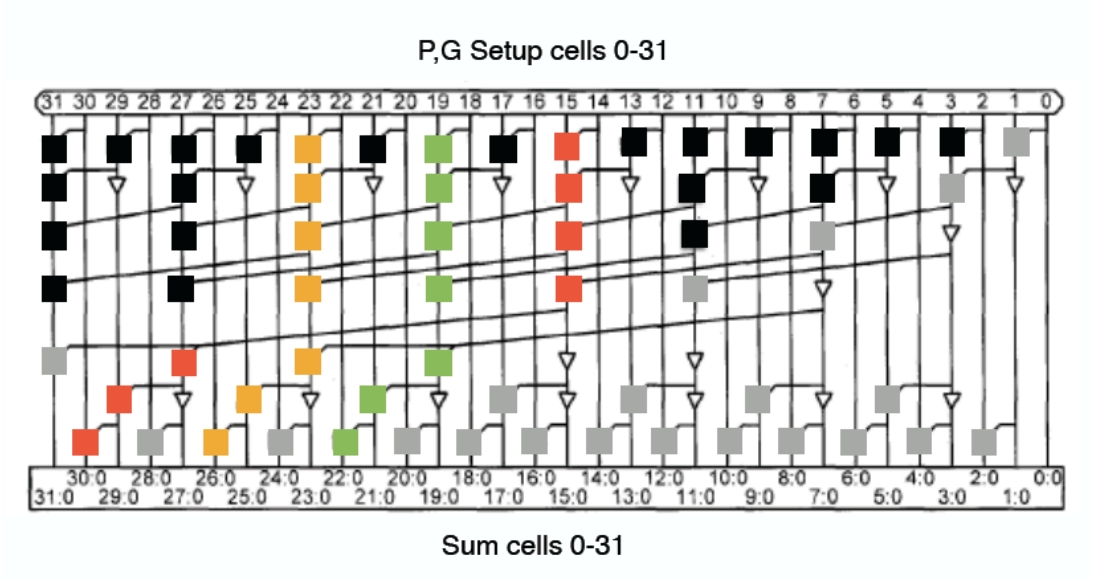


Figure 14: The complete 32-bit adder with one path colored for each of the three outputs that has the longest path through the tree: 30, 26 and 22

Solution 7: Switching voltage and activity factor

a) In a 3-input NAND gate with proper scaling for equal worst-case resistance, the three nMOS transistors in series are scaled to have a resistance R/3 each while each of the three pMOS transistors in parallel has a resistance R. Thus, when all three inputs are connected together the n-net has a resistance R while the p-net has a resistance R/3 due to the three pMOS transistors R being connected in parallel. The k values are the inverse of the effective resistance values. So the resulting values are $k_p = 3$ and $k_n = 1$. The expression for the switching voltage is:

$$V_{SW} = V_{TN} + \frac{\Delta V}{1 + \sqrt{x}}$$

where we have $x = \frac{k_n}{k_p}$ and $\Delta V = V_{DD} - V_{TN} + V_{TP}$. So, in this particular case we have $x = 1/3$ and thus we have:

$$V_{SW} = V_{TN} + \frac{\Delta V}{1 + \sqrt{1/3}}$$

Which we can also evaluate as:

$$V_{SW} = V_{TN} + 0.63\Delta V$$

Thus, we see that the switching voltage will be a bit higher than the one we get with equal k values. This makes sense since in this case the pMOS k values is higher.

b) The activity factor is the probablity that a signal goes from 1 to 0. It can be expressed as:

$$\alpha = P_1(1 - P_1) = P_1 P_0$$

where P_1 is the probablity that the node is 1. This is also often written as

$$\alpha = P(1 - P) = P\bar{P}$$

where P is the probability of the signal being 1. An AND-OR-INVERT 2+1 gate consists of first a 2-input AND gate followed by a 2-input NOR gate where one of the inputs comes from the AND gate. At the output of a two-input AND gate we have

$$P_{\text{AND}} = P_A \times P_B \tag{15}$$

and for a 2-input NOR gate we have

$$P_{\text{NOR}} = \bar{P_A} \times \bar{P_B} \tag{16}$$

By combining (15) and (16)we find the expression for the probability at the output of the AOI gate

$$P_{\mathrm{AOI21}} = (1 - P_A \times P_B) \times \bar{P_C}$$

With $P_A = P_B = P_C = 0.5$ we have $P_{\mathrm{AOI21}} = 3/8$ and $\alpha_{\mathrm{AOI21}} = 15/64$.