Problem Set #8

The **due date** for this homework is **Tue 7 May 2013 12:59 PM EDT -0400**.

If you find anything unclear to you, save your answer (do *not* submit), and kindly let us know in the forum so that we can clarify if there is indeed ambiguity.

☐ **In accordance with the Coursera Honor Code, I (Matthew Kramer) certify that the answers here are my own work.**

Question 1

Static Timing Analysis: Basics

Which of these are correct statements about Static Timing Analysis?

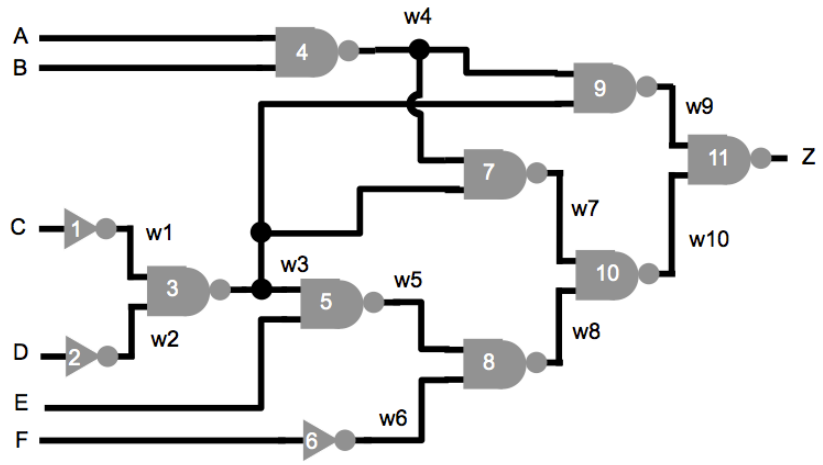
- ☐ Arrival time (AT) at node n means the earliest time the signal can become stable at node n.
- ☐ In the real world, the gate delay can be very complex. The factors that affect delay include gate type, gate loading, waveform shape, transition direction etc.
- ☐ When there is a positive slack at a node, it is possible to increase the delay at the node without degrading overall performance.
- ☐ Required arrival time (RAT) at a node refers to the latest time the signal is allowed to become stable at the node.
- ☐ In static timing analysis, we only care about the topological graph structure of the circuit, and we disregard the Boolean functionality of the circuit.
- ☐ The topological longest path in a circuit is a path where each node has the same maximum slack.
- ☐ All nodes on the longest delay path from source to sink will have the same worst case slack value.
- ☐ To find the worst paths in decreasing delay order, we can use an algorithm that is similar to Maze routing by storing the partial paths in a heap, where the item with most positive slack is always on top.

- A positive slack at a node means the node does not meet timing requirement, so it is always bad to see positive slack.
- For a 2-input NAND gate, the two gate inputs will have the same delay to the output because they are logically symmetric.

Question 2

STA Delay Graph, ATs, RATs and Slacks

Consider the small logic network shown below. There are 6 primary inputs (PIs) labeled: **A, B, C, D, E, F**. There is 1 primary output (PO): **Z**. There are 11 logic gates. There are 10 internal wires that represent gate outputs/inputs, labeled: **w1, w2, ..., w10**. The gate delay (cell arc) for each inverter is 2. The gate delay for each NAND2 is 3. Ignore delay for the wires. This logic operates on a clock with a **Cycle Time = 15**.



Do the following:

- Build the **Delay Graph** for this logic network, including one source (SRC) and one sink (SNK) node, and appropriate edges representing all the delays.
- Walking this graph from SRC to SNK, calculate the **Arrival Time (AT)** for each node in this graph. Just use your eyes to determine the necessary longest paths.
- Walking this graph from SNK to SRC, calculate the **Required Arrival Time (RAT)** for each node in this graph. Again, just use your eyes to determine the necessary longest paths.
- Using these computed ATs and RATs, compute the **Slack** value for each node in this graph.

Which of the following are correct statements of the results of this Static Timing

Analysis:

- [ ] The sink node has 2 edges entering it.
- [ ] After the static timing analysis, node w7 is labeled with AT=8, RAT=9, Slack=1.
- [ ] After the static timing analysis, node w1 is labeled with AT=1, RAT=2, Slack=1.
- [ ] There is an edge with weight 2 from node F to node w6 in this graph.
- [ ] The path SRC, C, w1, w3, w5, w8, w10, Z, SNK is a worst-case longest-delay path, with slack = -1 on each node.
- [ ] There is an edge with weight 3 from node w10 to node SNK in this graph.
- [ ] After the static timing analysis, node w6 is labeled with AT=2, RAT=6, Slack=4.
- [ ] The path SRC, D, w2, w3, w5, w8, w10, Z, SNK is a worst-case longest-delay path, with slack = 0 on each node.
- [ ] The delay graph has 19 nodes including the source and the target nodes.
- [ ] The path SRC, D, w2, w3, w5, w8, w10, Z, SNK is a worst-case longest-delay path, with slack = -2 on each node.

Question 3

Elmore Delay Analysis: Basics

Which of these are correct statements about interconnect (wire) delay in general, and the Elmore Delay Model in particular?

- [ ] According to Elmore delay, the delay between the root node and a leaf node in an RC tree equals to the root node resistance times the downstream capacitance.
- [ ] A wire segment can be modeled using a Pi model, which symmetrically splits capacitance into two halves between the two ends of the equivalent resistor of the segment.
- [ ] Elmore delay is the most precise and accurate measure of the electrical delay through an RC network. Thus it is very popular.
- [ ] The resistance of a metal wire segment is proportional to its length and inversely proportional to its width.

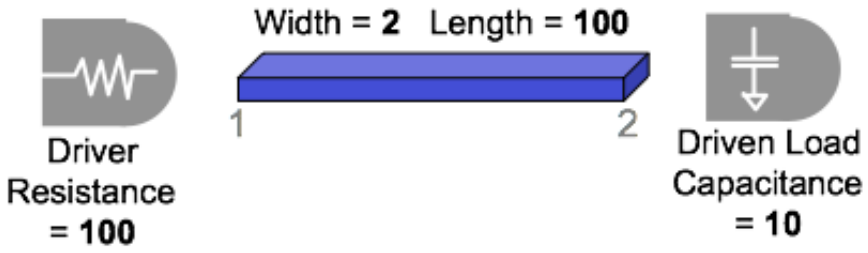
- Downstream capacitor of a resistor means all of the capacitors that are connected to the same circuit node right after this resistor.
- Parallel capacitors (i.e., several capacitors each hanging off the same electrical node) can be replaced by one bigger capacitor whose capacitance is the sum of all the individual capacitors.
- If the height and length are fixed for a given metal wire segment, the resistance will be decreased when its width is smaller.
- An RC tree is a tree of capacitors, with resistors hanging at all intermediate tree nodes.
- In an RC tree, the root is where signal is input, and the leaves are the driven outputs.
- Elmore delay is a simple, electrically based estimator for the actual wire delay; there are more sophisticated electrical models that can be used when we need to handle more complex electrical effects.

Question 4

Elmore Delay: Bad, Bad Elmore!

Your office-mate at work is a very nice person, but is not really very clever about the Elmore Delay. Your boss has assigned this person a simple analysis problem, shown below: one driver gate, one wire, and one driven load gate. Your boss asks: **What is the delay**? The electrical parameters are as follows: (don't worry about the units)

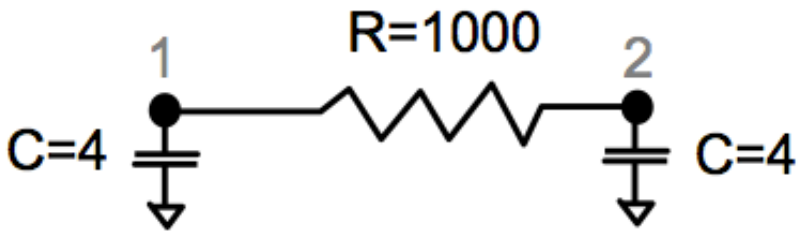
- Driver resistance: **100**
- Driven load capacitance: **10**
- Pi-model resistance formula: **R = 10 L/W**
- Pi-model capacitance formula: **C = (0.02) LW**



(Two electrical nodes labeled 1, 2, for visually impaired participants.

Your co-worker constructs the RC tree shown below. And computes the delay as

1000*4 = 4000. Your boss is very, very unhappy; this is very, very wrong.



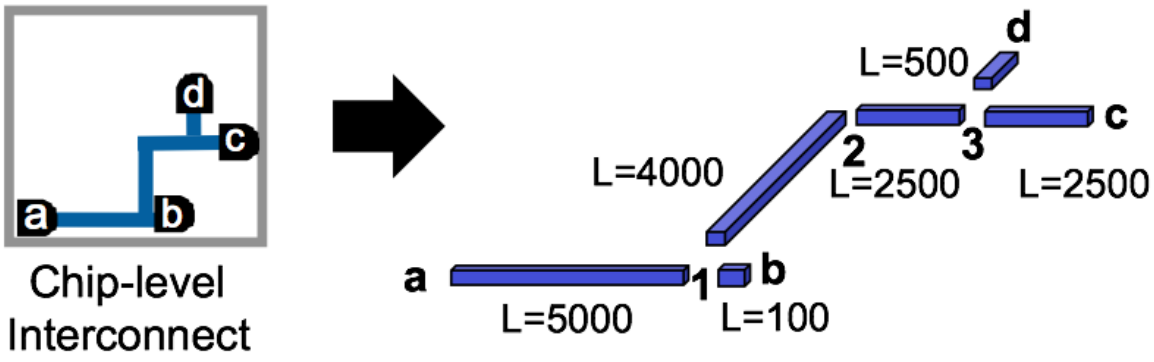
Which of the following are statements you should tell your co-worker, to instruct them on what they did incorrectly in this analysis?

- [ ] You forgot to include the driver resistance at the root of the tree. So, there should be another resistor with value R=100 connecting into node 1.
- [ ] The C value for the pi model is indeed 4 = (0.02)*2*100. But you dont put all that value on each capacitor in the pi model: you put half on each capacitor. So the C value for the capacitor at node 1 should be 2, not 4.
- [ ] You use the wrong capacitor value at the output; you forgot to divide the segment C value in half, and you also forgot to add in the value of the load capacitor at node 2; this C value should be 2+10 = 12.
- [ ] Hey! You need to add the driver resistance to the wire resistance, which adds up to 1100. Thus the correct value of delay will be 1100*14 = 15400.
- [ ] The right answer (with the correct resistance, driver and load gates added) should be: 100*14 + 500*12 = 7400 for the total Elmore delay to the output gate.

Question 5

Elmore Delay: RC Tree Model

Consider the large, chip-level interconnect wire shown in the figure below. The wire is a 4-point net: gate a drives the net; gates **b**, **c** and **d** are driven by the net. The wire decomposes into 6 separate metal segments, with 3 internal nodes labeled 1 - 3 in the figure. The length of each wire (in microns = 10^{-6} meters) is labeled next to each wire.



For clarity, the table below gives the dimensions of each wire segment; all number are in microns:

Segment	Length L	Width W
a-1	5000	0.1
1-b	100	0.05
1-2	4000	0.1
2-3	2500	0.05
3-c	2500	0.05
3-d	500	0.05

To compute the Elmore delay, we assume the following electrical parameters:

- Resistance of driver gate **a**: **300**
- Capacitance of the driven gates **b, c, d**: **1 x 10^{-6}**
- Pi-model resistance calculation for a segment of Length **L** and Width **W**: **R = (0.5) L/W**
- Pi-model capacitance calculation for a segment of Length **L** and Width **W**: **C = (0.1 x 10^{-6}) W L**

The parameters are roughly (very roughly) what we would expect for a modern ASIC in a leading-edge nanometer-scale fabrication process. Don't worry about the units, if you follow the above numbers, the final delay numbers will be scaled to produce answers in **nanoseconds**.

Do this: build the pi-model for each of the segments in the wire, then build the RC tree model for this wire, including driver and driven gates. Lump the capacitors at each internal node into one **big capacitor**. Which of the following are correct statements about the model:

- [ ] The final RC tree has 14 capacitors hanging off the internal nodes.

- The pi model for segment 1-2 has resistance R = 20000, and two capacitors each with C = 20 x 10^{-6}.
- The pi model for segment a-1 has resistance R= 25000, and one capacitor with C= 25 x 10^{-6} and another capacitor with C = 45.25 x 10^{-6}.
- The pi model for segment 1-b has resistance R = 1000, and two capacitors each with C = 0.25 x 10^{-6}.
- The value of the capacitor hanging off node 2 in the tree is 26.25 x 10^{-6}.

Question 6

Elmore Delay: Evaluating an RC Tree

Consider again the chip-level interconnect example and electrical parameters from Question 5. Now, we want to use the RC tree model you built in that problem to evaluate the delays for this wire. Which of the following are correct statements about these delays:

- The delay from the driver gate at a to driven gate d is 4.0188 ns.
- The delay from the driver gate at a to driven gate c is 41.888 ns.
- If we increased the length of segment 1-2 in the net, the delays to c, d would increase, but the delay to b would also increase.
- The delay from the driver gate at a to driven gate d is 3.9435 ns.
- The delay from the driver gate at a to driven gate c is 4.1888 ns.

Question 7

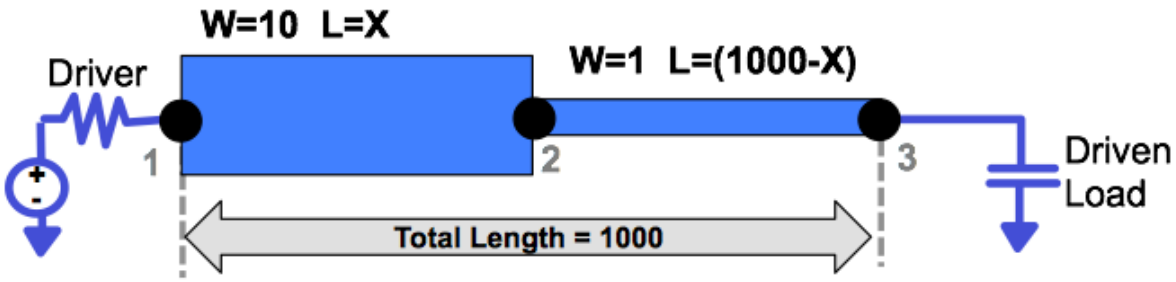
Elmore Delay: Wires as Parametric Objects

A highly useful feature of the Elmore delay model is it produces fairly simple algebraic equations that link the shape of a wire to its delay. So far, we have just assumed that the shape was pre-determined for example, by the router. But we can also use the Elmore delay for optimization: assume we want a certain behavior from the wire, then

solve for the shape that makes this happen. This is another really important use for the Elmore delay.

One good example is a **tapered wire**: the wire starts out wide, but ends narrow. The idea is that we control the shape to design a specific behavior we want. This is a common sort of optimization when the net is a large fanout interconnect like a clock, which connects to many storage elements. We want the segments near the root to be wide (so the resistance is low, and the overall delay is low; also for some other electrical reasons related to the need for this segment to handle a lot of current). But we do not need the segments near the leaves to be wide, and indeed, we make then narrow to reduce the overall capacitance load.

Lets do a small problem to illustrate. Consider the small parameterized wire shown below. There are two wire segments, and the total length of the wire is 1000. The wide segment at the start has length=X; the narrow segment at the end has length=(1000-X).



(Three electrical nodes labeled 1, 2, 3, for visually impaired participants. Wire with width 10 and length X from node 1 to 2; wire with width 1 and length (1000-X) from node 2 to 3)

Here are the necessary electrical parameters to compute the Elmore Delay:

- Driver resistance: **500**
- Load capacitance: **1×10^{-4}**
- Pi-model resistance calculation: **R = 500 L / W**
- Pi-model capacitance calculation: **$C = (2 \times 10^{-6})$ L W**

When we change the value of **X**, we change the taper of the wire, and thus the delay of the wire. We can design the shape of the wire to achieve the behavior we want.

Do this: solve for the correct value of **X** to make the delay at the load equal to **530**. Again, don't worry about the units. Type the number in the box. You can just type in

the **nearest integer**, from 0 to 1000.

Hint: You can do the math, and solve for the right answer. Its just algebra. Or, you can write the equation in a program, and then just try values until you find the right one. You can even do this in a spreadsheet and get the right answer. We dont care how you do it we just want the right answer. To assist on this, we will also tell you that the correct delay value when **X=900** is **437.65**.

☐ **In accordance with the Coursera Honor Code, I (Matthew Kramer) certify that the answers here are my own work.**

Submit Answers Save Answers

You cannot submit your work until you agree to the Honor Code. Thanks!