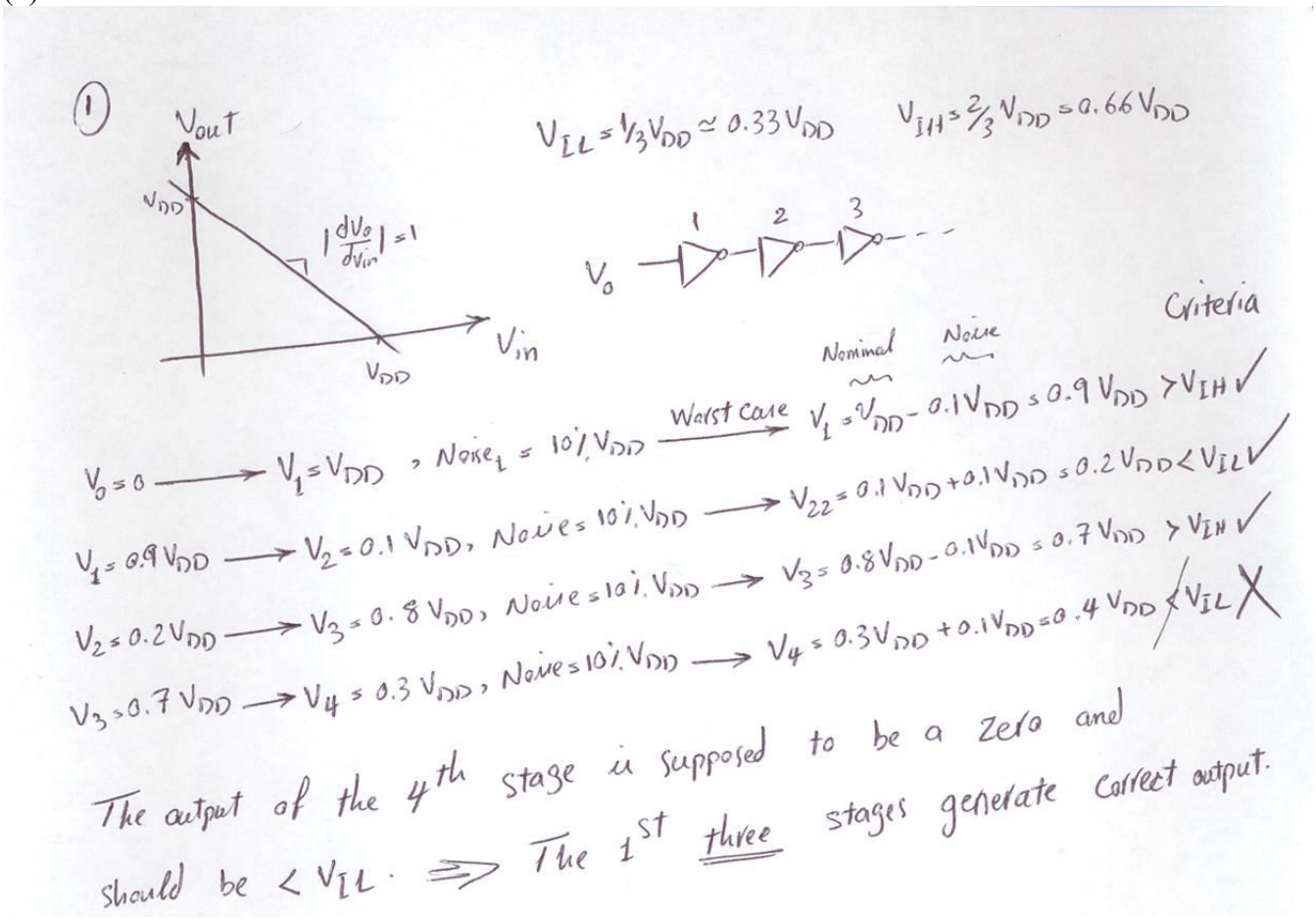
(1) P results more delay

(2) Ratio: PMOS vs NMOS = 6 vs 1
A = 3C B = 8C

(3)



4) The key concept for this problem is that restoration (or regeration) is necessary for correct operation in the presence of noise. Given that the transfer function is piecewise-linear, restoration requires that NM_H and NM_L each have non-zero width. This implies that $g > 1$. That is as far as we can go without knowing the amount of noise that must be tolerated. If we knew the required NM_H and NM_L, geometrical analyis leads to

$$g = \Delta y / \Delta x$$

$$g = V_{DD} / (V_{DD} - NM_H - NM_L).$$

5) MCM vs. single-chip ICs: MCMs can integrate different dies produced using different fabrication processes. However, they are more expensive than single-chip ICs. MCM interconnects have higher capacitance than on-die interconnect.

MCM vs. PCB: MCMs interconnects have lower capacitance than PCB interconnects, allowing increased frequency and reduced energy consumption. However, they may be more expensive.

6)
$\Phi = C_{load}/C_{init} = 83.7$
$\varphi_{opt} = e^{\gamma/\varphi opt+1}$
$\varphi_{opt} = e^{1.1/\varphi opt+1}$
Solving numerically yields $\varphi_{opt} = 3.67$.
$\varphi_{opt}^{n} = \Phi$
$3.67^{n} = 83.7$
n log(3.67) = log(83.7)
n = log(83.7) / log(3.67) = 3.4
Thus, the optimal number of stages is 3 or 4. However, we know that the output signal must not be inverted, leaving 4 the only option. Given the use of four stages, $\varphi = \Phi^{1/4} = 3.02$. This yields the following taper in widths: 4λ, 12.1λ, 36.6λ, 110.7λ. We can sanity check this by multiplying 110.7λ by 3.02 to get 334.8 (which is four times C_{load}). Now we need to discretize. If we were pushing very hard for optimality, we would need to consider all combinations of [4λ], [12λ, 13λ], [36λ, 37λ], and [110λ, 111λ]. However, we will come quite close to optimal by rounding. This yields 4λ, 12λ, 37λ, and 111λ.

7)

The smallest inverter has a gate capacitance equal to the final load. Thus, it will take the first inverter as long to charge the inserted inverter as it would to drive the load directly. So, no additional inverters should be added.

8)

(a) Dynamic power consumption

35–55%

(b) Short-circuit power consumption

8–10%

(c) Drain junction leakage power consumption

1%

(d) Gate leakage power consumption

3–5%

(e) Subthreshold leakage power consumption

30–50%

(9)

7)a) Could use only Vdd or gnd on MOSFET sources. Better: Use MUX-based design style. Need buffer or inverter on output to restore. Inverters more efficient in CMOS, so start from $\bar{f}$.

$$f(a,b,c) = ab + \bar{a}\bar{b}\bar{c}$$

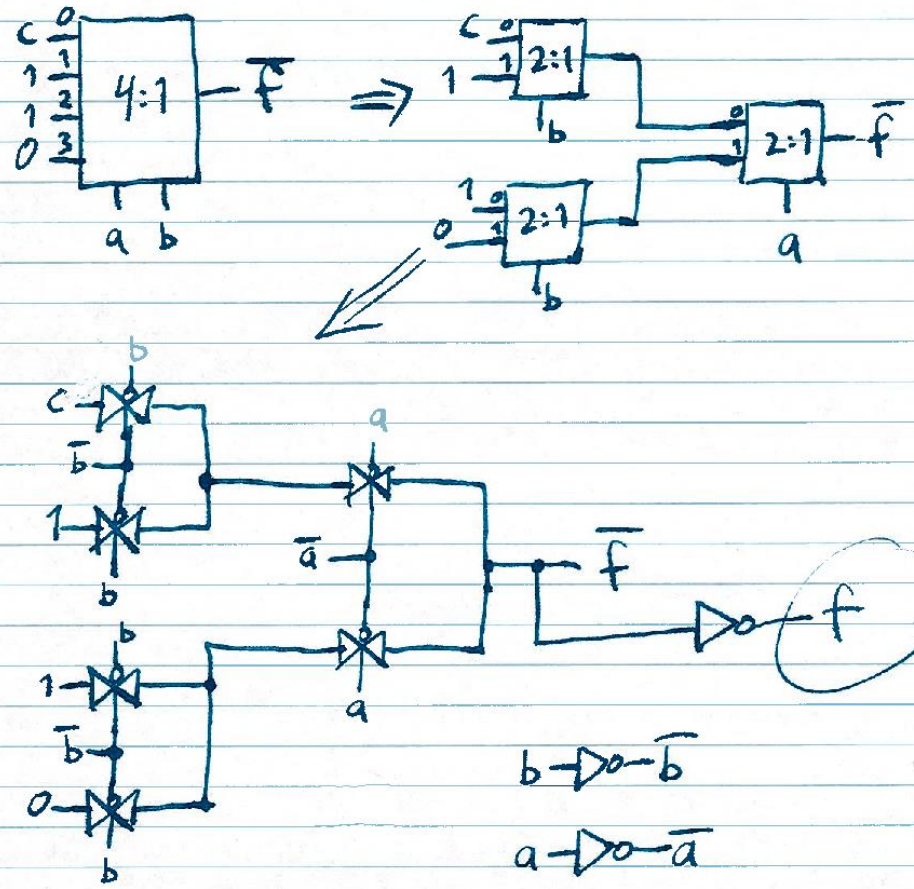
$$\bar{f}(a,b,c) = \overline{ab + \bar{a}\bar{b}\bar{c}}$$

$$= \overline{ab} \cdot \overline{\bar{a}\bar{b}\bar{c}}$$

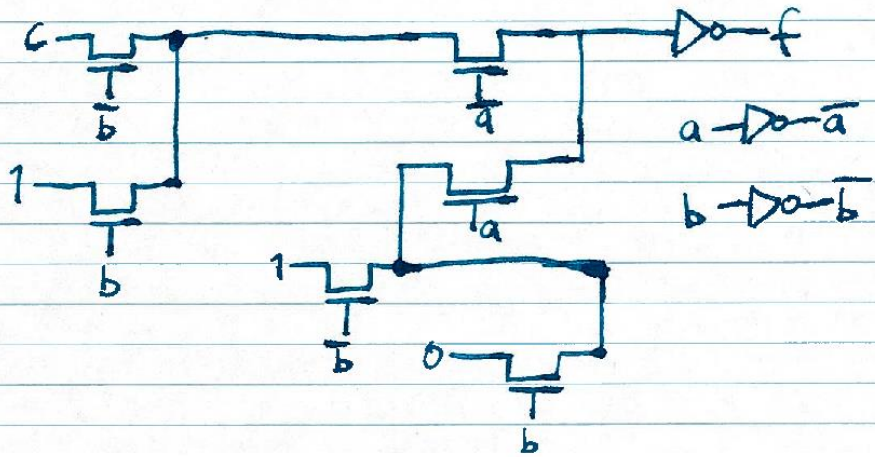
$$= (\bar{a}+\bar{b})(a+b+c)$$

a	b	c	$(\bar{a}+\bar{b})(a+b+c)$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

7)a) cont.)



7)b) Starting from soln. to 7)a).



10) a)

out = (a' + b')(c' + d'g') + e'f'(g' + c'd')

10) b)

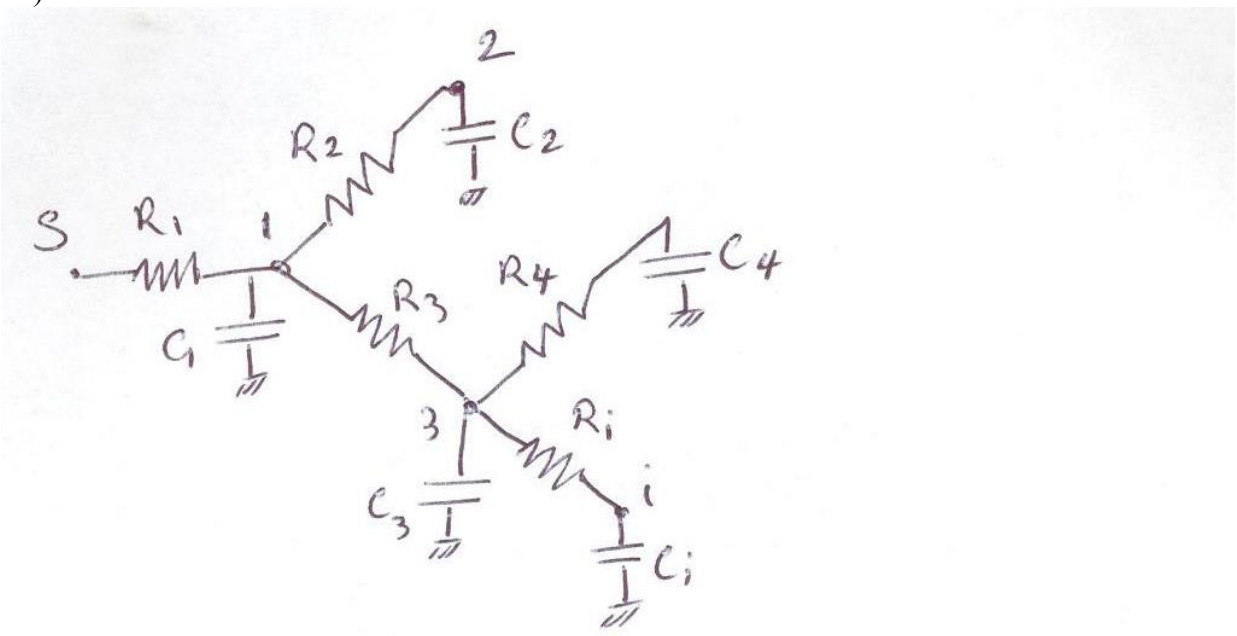
If “complex calculations” were required, we would need to take into account the impact of junction capacitances and input gate capacitances on delay. If you did this, good job. However, the delay can be approximated by considering the resistances of the pull-up and pull-down networks. First, we need to determine the inputs resulting in the worst-case on resistances for the pull-up and pull-down networks.

Pull-up: Based on PMOSFET widths, the worst-case on path is a'd'g' or b'd'g', with an effective width of 8/5um. All other inputs must be high. We need a transition that ends in this case and starts with the pull-up network off. a'bcdg' → a'bcd'g' is one valid answer.

Pull-down: One worst-case on-path is abdg with other inputs low. There are other worst-case on-paths. We need a transition that ends in this case and starts with the pull-down network off. abc'd'e'f'g→ abc'de'f'g works.

11) Because the wire is too long, the lumped model is no longer accurate. Instead, we will use the Elmore Model

12)



$$\tilde{C}_{i2} = R_i C_3 + R_i C_4 + (R_i + R_3) C_1 + (R_i + R_3 + R_2) C_2$$