3e3.1 3.201 The "probably" cases may cause damage to the gate if sustained.

(a) 0 (b) 1 (c) 0 (d) undefined
(e) 1 (f) probably 1 (g) probably 0 (h) probably 0

3e3.15 3.206 Simple, inverting CMOS gates generally have two transistors per input. Four examples that meet the requirements of the problem are 4-input NAND , 4-input NOR , 2-in, 2-wide AND–OR–INVERT , and 2-in, 2-wide OR–AND–INVERT.

3e3.22 3.208 Current is positive if it flows *into* a node. Therefore, an output with negative current is *sourcing* current.

3e3.23 3.209 The 74HC00 output drive is so weak, it's not good for driving much:

(a) Assume that in the LOW state the output pulls down to 0.33V (the maximum V_{OL} spec). Then the output current is $(5.0\text{V})/120\Omega = 41.7\text{mA}$, which is way more than the 4-mA commercial spec.

(b) For this problem, you first have to find the Thévenin equivalent of the load, or 148.5Ω in series with 2.25V. In the HIGH state, the gate must pull the output up to 3.84V, a difference of 1.59V across 148.5Ω, requiring 10.7 mA, which is out of spec. In the LOW state, we have a voltage drop of $2.25\text{V} - 0.33\text{V}$ across 148.5Ω, so the output must sink 12.9 mA, again out of spec.

3e3.29 3.211 The purpose of decoupling capacitors is to provide the instantaneous power-supply current that is required during output transitions. Printed-circuit board traces have inductance, which acts as a barrier to current flow at high frequencies (fast transition rates). The farther the capacitor is from the device that needs decoupling, the larger is the instantaneous voltage drop across the connecting signal path, resulting in larger spike (up or down) in the device's power-supply voltage.

3e3.32 3.213 (a) 5 ns.

3e3.39 3.218 The resistor must drop $5.0 - 2.0 - 0.37 = 2.63\text{V}$ with 5mA of current through it. Therefore $r = 2.63/0.005 = 526\Omega$; a good standard value would be 510Ω.

3e3.49 3.222 For each interfacing situation, we compute the fanout in the LOW state by dividing I_{OLmax} of the driving gate by I_{ILmax} of the driven gate. Similarly, the fanout in the HIGH state is computed by dividing I_{OLmax} of the driving gate by I_{IHmax} of the driven gate. The overall fanout is the lower of these two results.

	Low-state		High-state		Overall	Excess	
Case	Ratio	Fanout	Ratio	Fanout	Fanout	State	Drive
74LS driving 74LS	$\frac{8\text{mA}}{0.4\text{mA}}$	20	$\frac{400\mu\text{A}}{20\mu\text{A}}$	20	20	none	
74LS driving 74S	$\frac{8\text{mA}}{2\text{mA}}$	4	$\frac{400\mu\text{A}}{50\mu\text{A}}$	8	4	HIGH	$200\mu\text{A}$

3e3.55 3.226

				LOW-state			HIGH-state		
R_{VCC} (Ω)	R_{GND} (Ω)	V_{Thev} (V)	R_{Thev} (Ω)	$V_{Thev} - V_{OL}$ (V)	I_{OL} (mA)	OK?	$V_{OH} - V_{Thev}$ (V)	I_{OH} (μA)	OK?
470	—	5.0	470	4.5	9.57	no	<0	—	yes
330	470	2.9375	193.875	2.4375	12.57	no	<0	—	yes

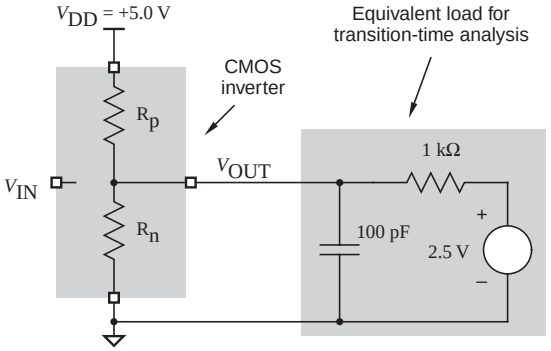
3e3.56 3.227

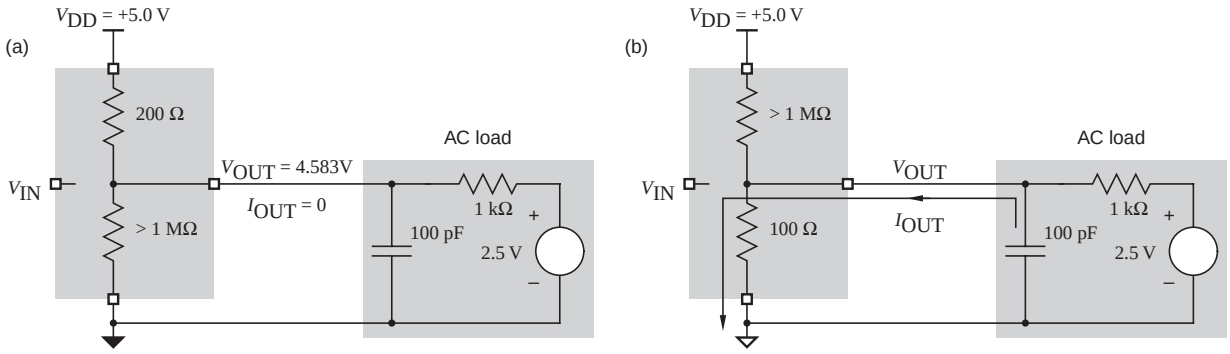
	LOW-state			HIGH-state		
Case	V_{ILmax}	$V_{OLmax(T)}$	Margin	V_{IHmin}	$V_{OHmin(T)}$	Margin
74HCT driving 74LS	0.8V	0.33V	0.47V	2.0V	3.84V	1.84V

3e3.57 **3.228** For each interfacing situation, we compute the fanout in the LOW state by dividing I_{OLmax} of the driving gate by I_{ILmax} of the driven gate. Similarly, the fanout in the HIGH state is computed by dividing I_{OHmax} of the driving gate by I_{ILmax} of the driven gate. The overall fanout is the lower of these two results.

	LOW-state		HIGH-state		Overall	Excess	
Case	Ratio	Fanout	Ratio	Fanout	Fanout	State	Drive
74HCT driving 74LS	$\frac{4\text{mA}}{0.4\text{mA}}$	10	$\frac{4000\ \mu\text{A}}{20\ \mu\text{A}}$	200	10	HIGH	3800 μA

3e3.68 **3.233** Including the DC load, a CMOS output's rise and fall times can be analyzed using the equivalent circuit shown to the right. This problem analyzes the fall time. Part (a) of the figure below shows the electrical conditions in the circuit when the output is in a steady HIGH state. Note that two resistors form a voltage divider, so the HIGH output is 4.583 V, not quite 5.0 V as it was in Section 3.6.1. At time $t = 0$ the CMOS output changes to the LOW state, resulting in the situation depicted in (b). The output will eventually reach a steady LOW voltage of 0.227 V, again determined by a voltage divider.





At time $t = 0$, V_{OUT} is still 4.583 V, but the Thévenin equivalent of the voltage source and the two resistors in the LOW state is 90.9 Ω in series with a 0.227-V voltage source. At time $t = \infty$, the capacitor will be discharged to the Thévenin-equivalent voltage and V_{OUT} will be 0.227 V. In between, the value of V_{OUT} is governed by an exponential law:

$$\begin{aligned} V_{OUT} &= 0.227\text{V} + (4.583 - 0.227\text{V}) \cdot e^{-t/(R_n C_L)} \\ &= 4.356 \cdot e^{-t/(90.9 \cdot 100 \cdot 10^{-12})}\text{V} \\ &= 4.356 \cdot e^{(-t)/(90.9 \cdot 10^{-9})}\text{V} \end{aligned}$$

Because of the DC load resistance, the time constant is a little shorter than it was in Section 3.6.1, at 9.09 ns. To obtain the fall time, we must solve the preceding equation for $V_{OUT} = 3.5$ and $V_{OUT} = 1.5$, yielding

$$t = -9.09 \cdot 10^{-9} \cdot \ln\frac{V_{OUT}}{4.356}$$

$$\begin{aligned} t_{3.5} &= 1.99\text{ ns} \\ t_{1.5} &= 9.69\text{ ns} \end{aligned}$$

The fall time t_f is the difference between these two numbers, or 7.7 ns. This is slightly shorter than the 8.5 ns result in Section 3.6.1 because of the slightly shorter time constant.

3e3.85 3.236

```
/* Transistor parameters */
#define DIODEDROP 0.6   /* volts */
#define BETA 10;
#define VCE_SAT 0.2     /* volts */
#define RCE_SAT 50      /* ohms  */
#define MAX_LEAK 0.00001 /* amperes */

main()
{
    float Vcc, Vin, R1, R2;  /* circuit parameters */
    float Ib, Ic, Vce;       /* circuit conditions */

    if (Vin < DIODEDROP) {   /* cut off */
       Ib  = 0.0;
       Ic  = Vcc/R2;  /* Tentative leakage current, limited by large R2 */
       if (Ic > MAX_LEAK) Ic = MAX_LEAK;  /* Limited by transistor */
       Vce = Vcc - (Ic * R2);
    }
    else {                   /* active or saturated */
        Ib = (Vin - DIODEDROP) / R1;
        if ((Vcc - ((BETA * Ib) * R2)) >= VCE_SAT) {   /* active */
            Ic  = BETA * Ib;
            Vce = Vcc - (Ic * R2);
        }
        else {               /* saturated */
           Vce = VCE_SAT;
           Ic  = (Vcc - Vce) / (R2 + RCE_SAT);
        }
    }
}
```