DEPARTMENT OF ELECTRONIC AND ELECTRICAL ENGINEERING

Spring Semester 2011-12 (2 hours)

Answers to EEE310/EEE6036

1. a. *The pull-down network for a CMOS digital gate is as shown in Figure 1.*

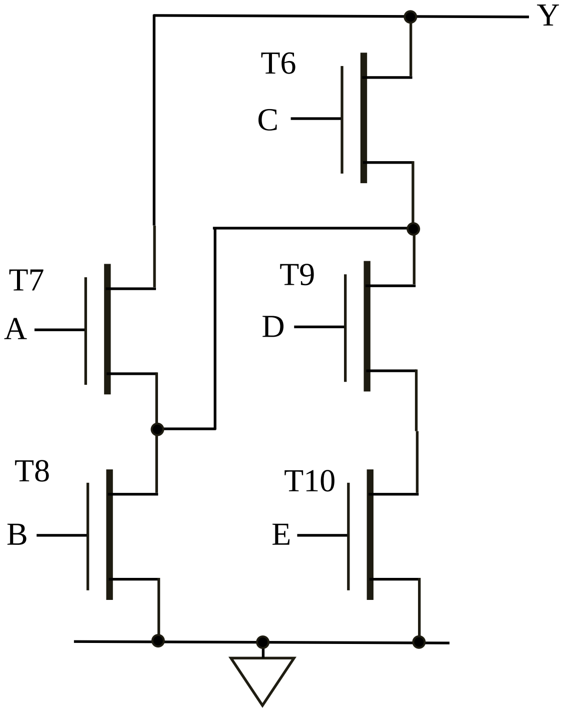
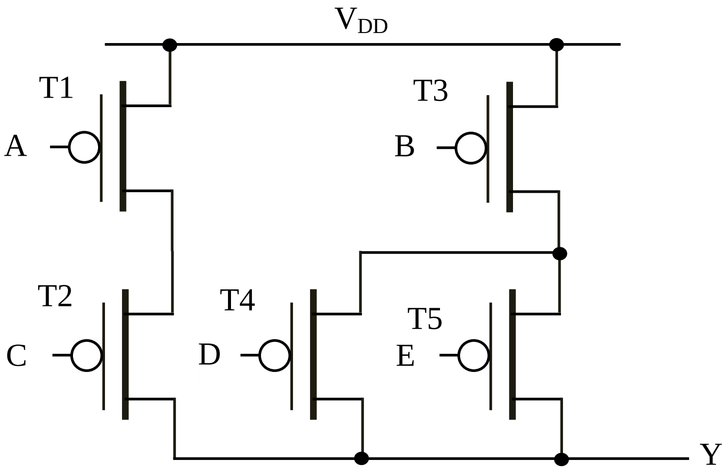


Figure 1: CMOS Pull-Down Network

i) *Draw the corresponding pull-up network for the gate;*



(5)

ii) *Determine the logical function of the gate;*

Looking at the pull-down network, $\bar{Y} = (A + C)(B + DE)$

Therefore,

$$Y = \overline{(A + C)(B + DE)} = \overline{A + C} + \overline{B + DE} = \bar{A}\bar{C} + \bar{B}\overline{DE} = \bar{A}\bar{C} + \bar{B}\bar{D} + \bar{B}\bar{E} \qquad \textbf{(5)}$$

iii) *Size the transistors in the gate, making the normal assumptions of a minimum-sized gate and that $\mu_H = 0.5\mu_E$;*

Labelling the transistors as shown above, T1=T2=T3=T4=T5=4, T6=T9=T10=3, T7=T8=2. **(5)**

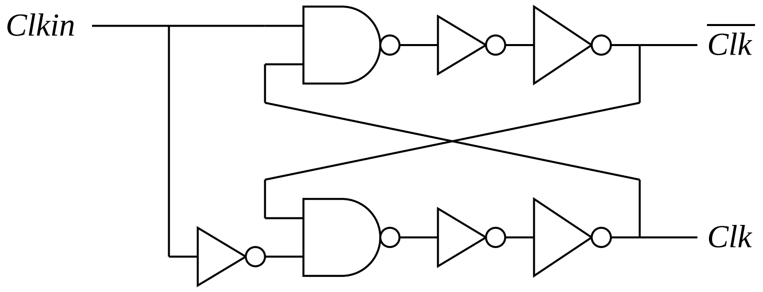
iv) *The substrate connections for the transistors in Figure 1 are not shown – where is it assumed that they are connected and why is this the case?*

Typically, you might expect the substrate to be connected to the sources of the transistors. However, all of these transistors sit in a common area of p-type silicon and so there is only one defined substrate voltage for the transistors and this will be the most negative voltage in the circuit V_{SS}. **(5)**

2. **a.** **i)** *Explain why might it be important to have a 2-phase non-overlapping clock generator in a clocked system?*

Generally speaking, flip-flops are constructed as master-slave arrangements where the master latch captures the value in the high part of the clock, going transparent in the low part of the clock cycle whilst the following slave part is transparent during the high part of the clock (passing the value from the master to the output) but latches this value in the low part of the clock cycle. This requires two, separate non-overlapping clock phases to ensure that both latches are never transparent at the same time. **(2)**

ii) *Show how such a clock generator could be constructed;*



(4)

b. **i)** *Describe, briefly, how a Delay-Locked Loop (DLL) operates;*

A DLL is normally constructed as a chain of N inverters whose individual delay can be controlled (by varying the supply voltage, for example). A clock signal is applied to the chain of inverters and emerges at the output delayed by a certain amount of time. This output is served against the input to the chain to minimise the phase difference. That is by XORing the input and output together and low-pass filtering the output, a phase detector can be constructed. The phase error is used to control the supply voltage to servo the phase error to a minimum – at which point the delayed clock will be in phase with the input clock. By selecting an output from part way along the chain of inverters, the output will have a well defined phase relationship with the input clock. That is, the N/2th inverter will be 180 degrees out of phase, the N/4th inverter will be 90 degrees out of phase. **(2)**

ii) *Give one use of a DLL;*

There are many uses. Typically a DLL can be used to adjust the phase of the clock coming on to the IC so that that internal clock is out of phase. This allows the on/off IC times to be accounted for so that the phase relationship between I/O signals at the pins – relative to the clock at the pin can be set i.e. setup and delay times. Alternatively, by XORing a 90 degree out of phase signal with the clock then a 2x clock frequency could be generated – phase locked to the input clock. **(2)**

c. *A signal is to be transferred between two clock domains. The receiving clock domain is running at a clock frequency of 1.2GHz whilst the sending clock domain uses a frequency of 350MHz. In both domains, the Flip-Flops have $T_0 = t_c = 0.05$ns.*

i) *Show that the effect of using a simple Flip-Flop as a synchroniser would* *result in circa 0.6 upsets/second*

The standard equation for calculating upsets is $upsets = T_0 e^{-\frac{t_o}{t_c}} f_{data} f_{clk}$ and assuming that the data frequency is 175MHz (half the clock frequency in the sending domain – it can be no greater than this) and that the observation time, t_o, is 1/1.2x10^{9} i.e the clock period in the receiving domain, this gives 0.05x10^{-9}*e$^{-16.6}$*1.2x10^{9}*175x10^{6} = 0.6066 upsets per second. **(2)**

ii) *It is estimated that a rate of upsets of fewer than 1x10^{-9} upsets/second is sufficient for the overall design. Design a synchroniser capable of meeting this requirement. What would the rate of upsets be for this synchroniser?*

Generally, adding flip flops in series, driven by the receiving clock domain reduces the upsets because for each flip flop added the observation time increases by the clock period. So, for one flip flop, the observation time is 0.83ns and 0.6 upsets per second are observed. For two flip flops the observation time is 1.66ns and the rate of upsets is 3.5x10^{-8} per second (which is still too high). If a third flip flop is added then this drops to 2x10^{-15} upsets per second, which is certainly low enough. **(4)**

iii) *What implications would the design of such a synchroniser have if the signal being passed from one domain to the other is acted upon in the receiving domain and a resultant signal is transferred back to the sending domain (e.g. a Request/Acknowledge handshake)?*

Clearly, the synchroniser allowing data to be transferred from the 1.2GHz domain to the 350MHz domain would be different. For one flip flop this would be 1.6x10^{-18} upsets per second (because t_o is much bigger). But this would mean that a signal passing back and forth would need three 1.2GHz clock cycles to be synchronised, at least one clock cycle to be processed within the domain, one 350MHz cycle to return to the first domain and then another 350MHz clock cycle for processing before it could return to the other domain. This could create problems for data processes co-operatively between domains or control signal (for a state machine in one domain with inputs/outputs to the other domain to another state machine, for example). **(4)**

3. **a.** *Develop a simple expression showing how power is dissipated in a digital circuit as a consequence of switched capacitance and extend this to show that the power dissipation of an entire IC is as shown in Eq 3.1:*

$$P = \alpha f_{clk} V_{DD}^2 \sum_{i=1}^{N} C_i \qquad \ldots 3.1$$

defining the meaning of all of the terms.

Substantially, the inputs to a gate and the interconnect between gates appears to be capacitive. As a wire connected to an input cycles from 0V→V_{DD}→0V, the capacitance is charged and discharged. The charge comes from V_{DD} and is discharged to 0V. This is a current.

Consider charging a capacitor, C to V_{DD}. The charge on the capacitor will be CV_{DD}. As the capacitor is discharged to 0V, this charge will flow down to earth.

The net charge moved through V_{DD} is, therefore, CV_{DD}. If this operation is being done f times a second then the charge moved per second is fCV_{DD}. and this is current, axiomatically. This current flows across V_{DD} and so the power dissipated by this switching activity, P, is fCV_{DD}^{2}. To put this in terms of a circuit, $C_{wire}=C_{in}+C_{int}$ (the sum of the gates' input capacitance to which the wire is connected and the capacitance of the wire), and f is the frequency at which the input is being driven.

If we are to extend this expression from a single gate to an entire circuit we must perform a summation across all of the gates and interconnect in the circuit. So if we assume that there are N wires in the design and the total capacitance associated with wire$_i$ and the load that it is driving is C_i then the total switching power dissipated by the circuit should be:

$$P = \sum_{i=1}^{N} f_i C_i V_{DD}^2$$

This expression assumes that each wire is switching at its own frequency, f_i. However, in practice, the switching of all the wires will be controlled by a single frequency f_{clk} and each wire will change state *on either* clock edge with a defined probability α_i. In this case, the expression becomes:

$$P = f_{clk} V_{DD}^2 \sum_{i=1}^{N} \alpha_i C_i$$

In many cases, it is possible to simplify the expression further by assuming a value for α that is representative for the whole circuit rather than an individual wire.

$$P = \alpha f_{clk} V_{DD}^2 \sum_{i=1}^{N} C_i$$

(6)

b. *In relation to reducing power dissipation, as technology scales down, explain what happens to terms in Eq 3.1, why these changes happen and the effect that these changes have on power dissipation.*

Technology scaling reduces dimensions and this can lead to a reduction in capacitance – hence power dissipation. However, wires getting closer together will tend to increase coupling capacitance and gate oxides getting thinner will increase capacitance. Moreover, to offset the effect of increasing resistance, wire cross sections are changing increasing the coupling between wires. However, as technology is scaled down, the IC remains the same size and so N is getting bigger – leading to an increase in power dissipation. Scaling V_{DD} down has always been the most effective way of reducing power dissipation but there are limits because current drive is proportional to (V_{DD}-V_T) and as V_{DD} reduces the current drive reduces to zero. f_{clk} increases as the transistors reduce in size and clearly this increases the power dissipated. **(6)**

c. *An IC is composed of 10M, 2-input NAND gate equivalents. The power supply voltage is 1.5V and the IC runs with an internal clock whose frequency is 3GHz. You can assume that a minimum-sized n-FET has a gate capacitance of 0.5fF and that μ_H = 0.5μ_E. You can also assume that the output of each NAND gate is connected to a wire with an average capacitance to ground equal to 3fF.*

Stating and justifying any other assumptions made, estimate the core power dissipation of the IC due to switched capacitance.

Each NAND gate input is connected to an n-FET that is 2x the minimum width and a p-FET that is 2x the minimum width i.e. the capacitance driven by each input is 4x0.5fF i.e. 2fF. Although the other end of the gate capacitance, the capacitance of the drains connected to the output should be considered and this is 0.5x1fF + 2 x 0.5x1fF i.e.1.5fF. Consequently, the total capacitance to be considered is the input, output and wiring capacitance i.e. 2+1.5+3=6.5fF. So for 10M gates this is 65nF. We can assume that α is around 0.1 for a control-dominated circuit. Thus, the power dissipation is:

$P = 0.1 \text{x} 1.5^2 \text{x} 3 \text{ x} 10^9 \text{x} 6.5 \text{x} 10^{-8} = 43.85\text{W}$ **(8)**

4. **a.** *A cell from a CMOS cell-library is laid-out as shown in Figure 4, below.*

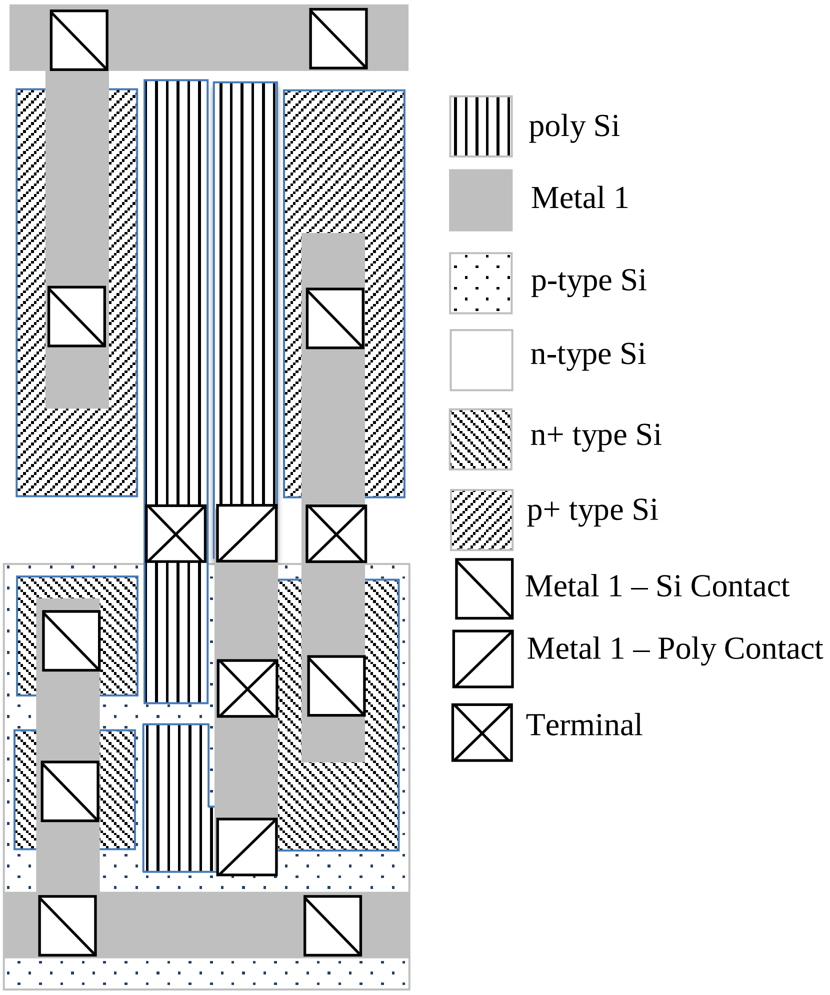
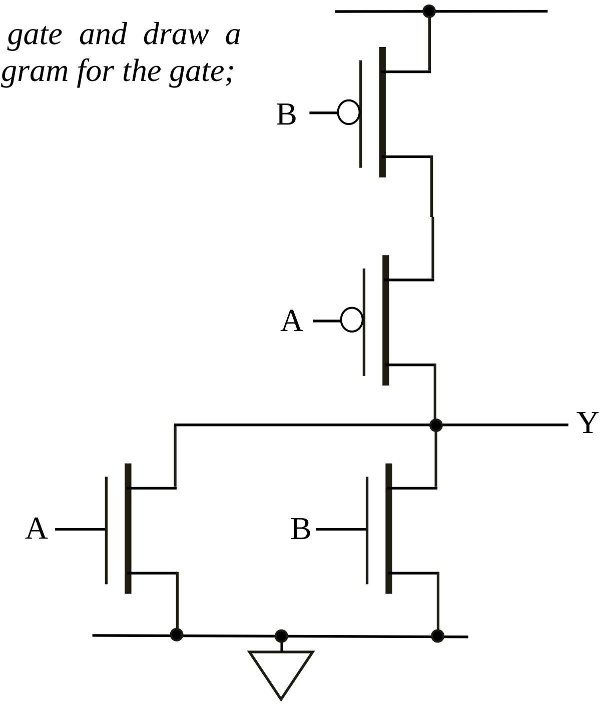


Figure 4: Layout

i) *Identify the function of the gate and draw a transistor-level schematic diagram for the gate;*

It is a 2-input NOR gate.



(8)

ii) *Estimate the ratio of μ_H to μ_E;*

For a minimum sized gate, assuming $\mu_H = \mu_E$, the n-FETS should be 1x width and the p-FETs should be 2x width. If $\mu_H <> \mu_E$ then the p-FETs should be 2 μ_E/μ_Hx Simple measurement shows that the p-FETs are approximately 4x wider than the n-FETs. Hence, $\mu_H=0.5\mu_E$. **(2)**

iii) *Identify the positive power supply (V_{DD}) and the negative power supply (V_{SS});*

The positive power supply should be connected to to the source of the top p-FET. Hence it must be the horizontal section of metallisation at the top of the figure. Conversely, the negative power supply will be the horizontal section of metallisation at the bottom of the figure. **(2)**

iv) W*hat is a self-aligned gate and how is it formed?*

In a real device, the poly gate on an n-FET will be doped n+ to reduce resistance and ensure that the threshold voltage is correct. The source and drain regions are also n+ and for control. The source and drain regions must start where the gate ends (laterally). In practice this can be simply arranged during fabrication. An area is opened (the active region) in the field oxide defining the area of the FET (the material below is p-type). A thin oxide is thermally grown over this area and the poly gate is then formed across the active region over the top of the thin (gate) oxide. Donor dopants are then implanted. Where the source and drain are the thin oxide allows the dopants to be implanted into the underlying silicon, forming n+ regions. The dopants are implanted into the poly gate (making it n+) but the gate acts as a mask preventing donor dopants being implanted into the channel below the gate. It is self-aligned because wherever the gate does not overlay the active area, source and drain regions are formed. **(4)**

v) *Is this an n-well or p-well process?*

The p-region underlying the n-FETs is delineated and, therefore, we can assume that it is a single, p-well process. **(2)**

vi) What is ‘field oxide’?

During fabrication, where the transistors are (the active area) there is bare silicon. Elsewhere, a thick oxide is grown, thermally, the field oxide. This passivates the surface and prevents implantation of the silicon below and, for example, the formation of other devices. **(2)**

NLS