PROBLEM 1: LOGIC STYLES

Problems 3)b) and 3)c) from the EE141 Fall07 Midterm #2 (available on the web)

Solution: See solutions posted on the web.

PROBLEM 2: FLIP-FLOP TIMING

Problem #4 from the EE141 Fall08 Final (available on the web).

Solution: See solutions posted on the web.

PROBLEM 3: “PULSED” LATCH TIMING

Consider the simple state machine shown above. A, B, and C represent combinational logic blocks with the following properties:

- \( t_{\text{logic,min}} = 200 \text{ ps} \); \( t_{\text{logic,max}} = 1 \text{ ns} \);
- \( t_{\text{logic,min}} = 300 \text{ ps} \); \( t_{\text{logic,max}} = 2 \text{ ns} \);
- \( t_{\text{logic,min}} = 100 \text{ ps} \); \( t_{\text{logic,max}} = 0.5 \text{ ns} \);

The L-units represent positive latches clocked by \( \phi \) (i.e., the latches are transparent when \( \phi \) is high). These latches have a setup time of 150 ps and a \( t_{\text{d-q}} \) delay of 250 ps (when the
latch is transparent). The clock to output delay $t_{\text{clk-q}}$ is 100 ps, and $t_{\text{hold}}$ is 100 ps. The clock $\phi$ has a period $T_{\text{clk}}$ and is high for a duration of $T_{\text{on}}$ – in other words, the duty cycle of the clock is $T_{\text{on}}/T_{\text{clk}}$.

a) Determine the conditions on the clock necessary to avoid the occurrence of hold-time violations.

Solution:

We want to ensure that there is no way for a new value at the output of a latch to race through the logic so quickly that the value of the latch changes again before the clock goes low. There are two possible paths for this to happen: $L_1 \rightarrow A \rightarrow C \rightarrow L_2$, and $L_2 \rightarrow B \rightarrow C \rightarrow L_2$; since $t_{\text{logic,minA}}$ is less than $t_{\text{logic,minB}}$, the first of these two paths is the most critical one. Therefore, the maximum $T_{\text{on}}$ we can use is set by:

$$t_{\text{clk-q}} + t_{\text{logic,minA}} + t_{\text{logic,minC}} > T_{\text{on}} + t_{\text{hold}}$$

$$T_{\text{on}} < 100\text{ps} + 200\text{ps} + 100\text{ps} - 100\text{ps}$$

$$\rightarrow T_{\text{on}} < 300\text{ps}$$

b) Determine the absolute minimum clock period for this circuit to work correctly as well as the maximum duty cycle.

Solution:

The two possible paths that set the minimum clock period are the same two paths that set the maximum $T_{\text{on}}$. However, this time, since $t_{\text{logic,maxB}} > t_{\text{logic,maxA}}$, it is the $L_2 \rightarrow B \rightarrow C \rightarrow L_2$ path that we need to consider. There are actually two possible scenarios that could set the minimum clock period:

1) The old data arrives right before $\phi$ goes high, and the new data must arrive $t_{\text{setup}}$ before the falling edge of $\phi$. In this case, $T_{\text{clk}}$ is set by:

$$t_{\text{clk-q}} + t_{\text{logic,maxB}} + t_{\text{logic,maxC}} + t_{\text{setup}} < T_{\text{clk}}$$

$$100\text{ps} + 2\text{ns} + 0.5\text{ns} + 150\text{ps} < T_{\text{clk}}$$

$$\rightarrow T_{\text{clk}} > 2.75\text{ns}$$

2) The old data arrives exactly $t_{\text{setup}}$ before $\phi$ goes low (i.e., while $L_2$ is transparent), and the new data must arrive at most one clock cycle later (so that in the worst case, the new data also arrives exactly $t_{\text{setup}}$ before $\phi$ goes low on the next clock cycle). In this case, $T_{\text{clk}}$ is set by:

$$t_{\text{d-q}} + t_{\text{logic,maxB}} + t_{\text{logic,maxC}} < T_{\text{clk}}$$

$$250\text{ps} + 2\text{ns} + 0.5\text{ns} < T_{\text{clk}}$$

$$\rightarrow T_{\text{clk}} > 2.75\text{ns}$$
In general, we would have to take the maximum of the results from cases 1) and 2) to find the minimum clock period, but since $t_{\text{clk-q}} + t_{\text{setup}} = t_{\text{d-q}}$ for these latches, both cases give the same result of $T_{\text{clk}} > 2.75\text{ns}$.

Combining this with the answer from a), the maximum duty cycle of the clock is $T_{\text{on, max}} / T_{\text{clk,min}} = 300\text{ps} / 2.75\text{ns} \approx 10.91\%$

c) Suppose that due to some sloppy clock-network routing, the clock signal at L1 arrives 100ps earlier than the clock signal at L2. Calculate the absolute minimum clock period for this circuit to work properly as well as the maximum duty cycle.

Solution:

Clearly, this clock skew only affects the L1 $\rightarrow$ A $\rightarrow$ C $\rightarrow$ L2 path. Since this path does not impact the minimum cycle time, $T_{\text{clk,min}}$ is unchanged, and remains 2.75ns.

It is exactly this path that determines the maximum $T_{\text{on}}$ however; since the clock to L1 arrives early, there is more time for the new value to race through logic block A and C and be latched by L2. So, $T_{\text{on}}$ is now set by:

$$-t_{\text{skew}} + t_{\text{clk-q}} + t_{\text{logic,minA}} + t_{\text{logic,minC}} > T_{\text{on}} + t_{\text{hold}}$$

$$T_{\text{on}} < -100\text{ps} + 100\text{ps} + 200\text{ps} + 100\text{ps} - 100\text{ps}$$

$$\Rightarrow T_{\text{on}} < 200\text{ps}$$

So, the maximum duty cycle is now $200\text{ps} / 2.75\text{ns} \approx 7.3\%$.

PROBLEM 4: LIMITED SWITCH DYNAMIC LOGIC
The circuit above (with relative transistor sizes annotated) is a Limited Switch Dynamic Logic (LSDL) NOR3 gate, which is essentially a dynamic gate followed by a latch.

a) What purpose are the shaded transistors serving?

Solution:

The shaded transistors act as keepers to maintain a static output. Without these transistors, the input of the inverter would be floating during the pre-charge phase.

b) Assuming no propagation delay, complete the following ideal timing diagram.

Solution:
c) Calculate the $t_{\text{setup}}$ of this gate in terms of $t_{\text{inv}}$. In other words, how long before the falling edge of the clock do the inputs have to be stable to ensure that the correct value is latched at the output? You can assume that $\gamma=1$.

**Solution:**

The circuit is composed of three stages: the first stage is a domino gate, the second stage is the inverter with keepers, and the third stage is the output inverter. For $t_{\text{setup}}$ we only need to consider the propagation delay through the first two stages. Once the output of the second stage is set, the clock can safely transition without the risk of losing the latched data.

The delay of the first two stages can be analyzed using either logical effort or simply an RC model. We will describe the method based on logical effort. First, let’s calculate the logical effort and worst-case parasitic delay of the first stage during the evaluation phase:

$$LE_{\text{NOR3}} = \frac{2}{3}$$
$$p_{\text{NOR3}} = \frac{7}{3}t_{\text{inv}}$$
$$f_{\text{NOR3}} = \frac{4}{2} = 2$$

So, the delay of the first stage is $(2/3*2+7/3)t_{\text{inv}} = (11/3)t_{\text{inv}}$

For the second stage, the input to the latch transitions from high to low during evaluation, so we should calculate the logical effort and parasitic delay of the gate when it is pulling the output high. Since $V_x$ is low when the second stage is pulling its output high, we don’t need to worry about the NMOS keeper. Thus:

$$LE_{\text{latch}} = \frac{4}{3}$$
$$p_{\text{latch}} = \frac{5}{3}t_{\text{inv}}$$
$$f_{\text{latch}} = \frac{3}{4}$$

So, the delay of the second stage is $(4/3*3/4+5/3)t_{\text{inv}} = (8/3)t_{\text{inv}}$

Hence, the setup time is: $(19/3)t_{\text{inv}}$

d) Assuming that $A=B=C=1$, compare the activity factor at node $Out$ to the activity factor at node $V_x$ (which would be the output of a standard dynamic logic gate). What can you infer about the dynamic power consumption of static gates being driven by this LSDL gate compared to gates driven by domino logic?

**Solution:**

In the LSDL gate with $A=B=C=1$, $V_x$ transitions every cycle. However, because of the latch, $Out$ will remain constant despite the transitions on $V_x$. Thus, in comparison to a standard domino gate, the number of transitions at the output of an LSDL gate is
significantly lower, leading to less data-dependent dynamic power consumption. Of course, this reduced activity factor at the output doesn’t come for free – the LSDL gate has higher clock loading due to the extra clock transistor in the latch.