CSIE 2344, Spring 2019 — Homework 3

Due May 6 (Monday) at Noon

There are 96 points in total. Points will be deducted if no appropriate intermediate step is provided.

When you submit your homework on Gradescope, please select the corresponding page(s) of each problem.

1 Hazards (16pts)

This question is the same as the first one in Discussion of Week 7. Consider the three-level NOR circuit below and find all static and dynamic hazards in this circuit.

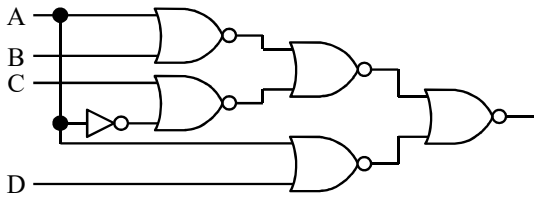


Figure 1: The three-level NOR circuit.

Answer: $F = (((A+B)'+(A'+C)')'+(A+D)')' = ((A+B)'+(A'+C)')(A+D) = (A'B'+AC')(A+D) = A'AB' + AC' + A'B'D + AC'D = A'AB' + AC' + A'B'D$. The corresponding Karnaugh map below has two implicants AC' and $A'B'D$, and we can find the following hazards.

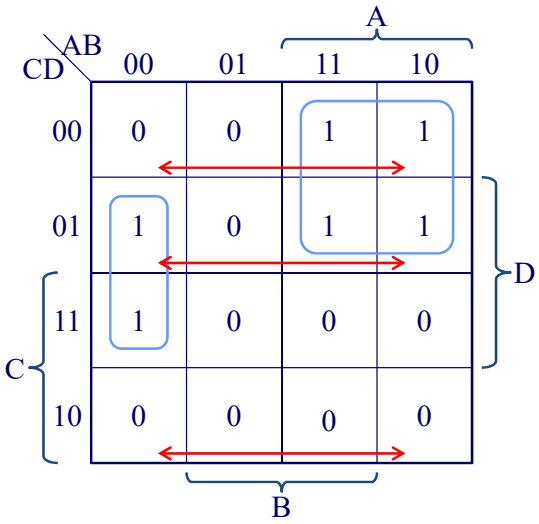


Figure 2: The Karnaugh map.

- Static-1 hazard: the transition between 0001 and 1001 as they are not covered by a prime implicant.
- Static-0 hazard: the transition between 0010 and 1010 as, from the term $A'AB'$, they have the same value 0 when $B' = 1$ and changing A.
- Dynamic hazard: the transition between 0000 and 1000 as, from the term $A'AB'$, they have different values when $B' = 1$ and changing A, and there are 3 paths propagating it. Note that 0011 and 1011 do not generate a dynamic hazard as there is only 1 path propagating it.

2 MUXes and Three-State Buffers (8pts)

Show how to make an 8-to-1 MUX using two 4-to-1 MUXes, two three-state buffers, and one inverter.

Answer:

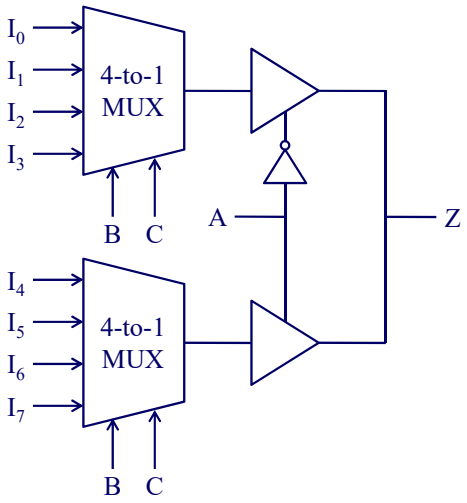


Figure 3: The implementation of an 8-to-1 MUX.

3 Latch (16pts)

A latch can be constructed from an OR gate, an AND gate, and an inverter connected as follows.

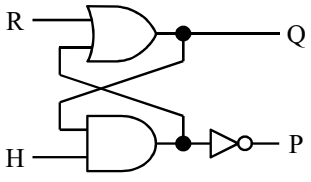


Figure 4: The latch.

1. What restriction must be placed on R and H so that P will always equal Q' (under steady-state conditions)?

2. Construct a next-state table and derive the characteristic (next-state) equation for the latch.

Answer:

1. P and Q must be different, so the restriction is that $R = 1$ and $H = 0$.
2. The next-state table is as follows:

R	H	Q	Q^+
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	X
1	0	1	X
1	1	0	1
1	1	1	1

By a Karnaugh map, $Q^+ = R + HQ$.

4 Lab 1 — Source Code (24pts)

1. (16pts) Print out your `cla_gl` source code and attach it with your Homework 3.
2. (8pts) Print out your `adder_rtl` source code and attach it with your Homework 3.

We may ask you to demo on May 13 or Jun 3. Points will be deducted if your demo fails. ***You can only demo with the source code same as that you submit***.

Answer: Example codes are below:

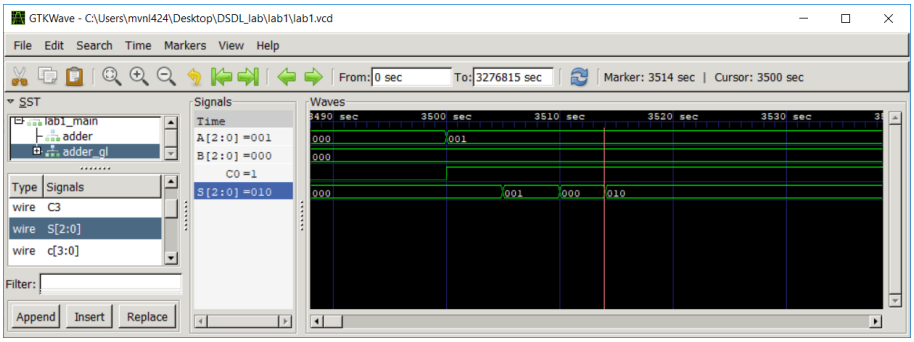
```
module cla_gl(
        output C3,
        output[2:0] S,
        input[2:0] A, B,
        input C0
        );
        wire[2:0] p, g;
        AND g0(g[0], A[0], B[0]);
        AND g1(g[1], A[1], B[1]);
        AND g2(g[2], A[2], B[2]);
        OR p0(p[0], A[0], B[0]);
        OR p1(p[1], A[1], B[1]);
        OR p2(p[2], A[2], B[2]);
        wire[3:0] c;
        assign c[0] = C0;
        assign C3 = c[3];
        wire w11, w21, w22, w31, w32, w33;
        AND c11(w11, c[0], p[0]);
        AND c21(w21, g[0], p[1]);
        AND4 c22(w22, c[0], p[0], p[1], 1);
        AND c31(w31, g[1], p[2]);
        AND4 c32(w32, g[0], p[1], p[2], 1);
        AND4 c33(w33, c[0], p[0], p[1], p[2]);
        OR c1(c[1], g[0], w11);
        OR4 c2(c[2], g[1], w21, w22, 0);
        OR4 c3(c[3], g[2], w31, w32, w33);
        wire[3:1] co;
        FA s0(co[1], S[0], A[0], B[0], c[0]);
        FA s1(co[2], S[1], A[1], B[1], c[1]);
        FA s2(co[3], S[2], A[2], B[2], c[2]);
endmodule
```

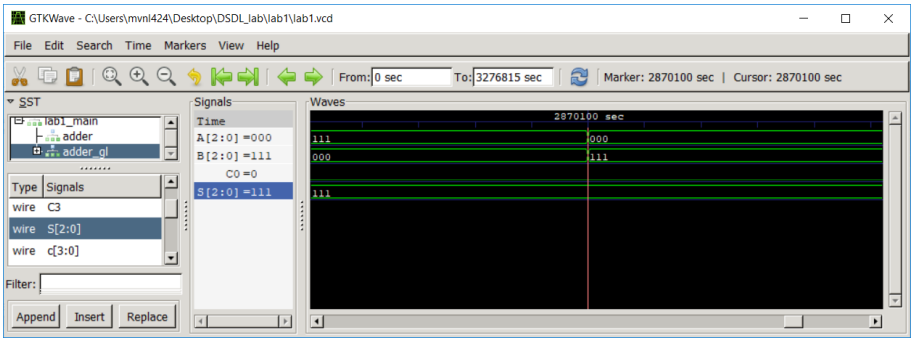
```
module adder_rtl(
        output C3,
        output[2:0] S,
        input[2:0] A, B,
        input C0
        );
        assign {C3, S} = A+B+C0;
endmodule
```

5 Lab 1 — Waveform (16pts)

1. Show the waveform of `cla_gl`.$S[2{:}0]$ on input transition from $000 + 000 + 0$ to $001 + 000 + 1$.
2. Show the waveform of `cla_gl`.$S[2{:}0]$ on input transition from $111 + 000 + 0$ to $000 + 111 + 0$.

Answer: The example waveforms are shown below.





6 Lab 1 — Propagation Delay (8pts)

Find the maximum propagation delay of `cla_gl` and one of the corresponding input transitions.

Answer: The maximum delay is 20 ticks. One input transition is from $000 + 000 + 0$ to $000 + 011 + 1$.

7 Lab 1 — Some Derivation (8pts)

Assume that only 2-input gates are used. Derive the number of levels needed in an n-bit carry-lookahead adder as a function of n.

Answer: C_n can be derived as follows

$$\begin{aligned}
C_1 &= G_0 + P_0 C_0, \\
C_2 &= G_1 + G_0 P_1 + C_0 P_0 P_1, \\
C_3 &= G_2 + G_1 P_2 + G_0 P_1 P_2 + C_0 P_0 P_1 P_2, \\
&\cdots \\
C_n &= \sum_{i=0}^{n-1} \left(G_i \prod_{j=i+1}^{n-1} P_j \right) + C_0 \prod_{j=0}^{n-1} P_j .
\end{aligned}$$

This is the summation of $n+1$ terms and the multiplication of at most $n+1$ terms, which implies $2\lceil \log_2(n+1) \rceil$ levels. Considering one level for P_i or G_i, the number of levels is $2\lceil \log_2(n+1) \rceil + 1$.

On the other hand, S_{n-1} is $C_{n-1} \oplus (A_{n-1} \oplus B_{n-1})$, which implies $2\lceil \log_2 n \rceil + 2$ levels.

Combining both, it needs $\max\{2\lceil \log_2(n+1) \rceil + 1, 2\lceil \log_2 n \rceil + 2\}$ levels.