Switching Circuits & Logic Design

Jie-Hong Roland Jiang
江介宏

Department of Electrical Engineering
National Taiwan University

Fall 2013

§8 Combinational Circuit Design and Simulation Using Gates

Melting Clocks, 1931
Salvador Dali (Spanish, 1904-1989)

http://arthistory.about.com/
Outline

- Gate delays and timing diagrams
- Hazards in combinational logic
- Simulation and testing of logic circuits

Design of Circuits with Limited Gate Fan-in

If a 2-level realization of a circuit requires more gate inputs than allowed, factoring the logic expression to obtain a multi-level realization may be necessary.

Example
- Realize $f(a,b,c,d) = \sum m(0, 3, 4, 5, 8, 9, 10, 14, 15)$ using 3-input NOR gates

```
\[
\begin{array}{c|cccc}
ab & 00 & 01 & 11 & 10 \\
\hline
cd & 00 & 0 & 0 & 0 \\
 01 & 0 & 0 & 0 & 0 \\
 11 & 0 & 0 & 0 & 0 \\
 10 & 0 & 0 & 0 & 0 \\
\end{array}
\]
```

\[
f' = b'd(a'c' + ac) + a'c(b + d') + abc'
\]

\[
f = [b + d' + (a + c)(a' + c')] [a + c' + b'd][a' + b' + c]
\]

Diagram of the circuit realization.
Gate Delays and Timing Diagrams

Gate Delays

- The output of a logic gate takes a finite time (**propagation delay**) to react to an input change
  - Propagation delays for IC gates are typically in a few nanoseconds ($ns = 10^{-9}$ sec)
  - Propagation delays for $0 \rightarrow 1$ and $1 \rightarrow 0$ output changes may be different

\[ X \rightarrow \quad X' \]

Voltage

\[ X \quad \rightarrow \quad X' \]

Time

\[ \epsilon_1 \quad \rightarrow \quad \epsilon_2 \]

Timing Diagrams

- A timing diagram shows various signals in the circuit as a function of time

\[ A \quad G_1 \quad G_2 \]

Assume each gate has a propagation delay 20 ns
Circuit with a delay element

Assume AND-gate has a propagation delay $\epsilon \mu s$

Hazards in Combinational Circuits

- When the input to a combinational circuit changes, unwanted switching transients (hazards) may appear in the output.
  - These transients occur when different paths from input to output have different propagation delays.

- Hazards are undesirable:
  - They consume power/energy
  - They may result in function errors in certain circuit design styles (e.g., domino logic)
  - They may slow down the performance of sequential circuits
Types of Hazards

- **Static 1-hazard**: a circuit output may momentarily go to 0 when it should remain a constant 1.
- **Static 0-hazard**: a circuit output may momentarily go to 1 when it should remain a constant 0.
- **Dynamic hazard**: a circuit output may change 3 or more times when it should change from 0 to 1 (or 1 to 0).

The steady-state output of the circuit is correct, but a switching transient appears at the output when the input is changed.

Hazards in 2-level Circuits

Hazards of 2-level AND-OR (OR-AND) circuits can be detected using K-maps and removed by adding terms (clauses):

- Static 1-hazard can appear in 2-level AND-OR circuits.
- Static 0-hazard can appear in 2-level OR-AND circuits.
Hazards in Combinational Circuits
Static 1-Hazard

Example
- Letting $A = C = 1$ and $B$ from 1 to 0 results in a static 1-hazard ($F = B + B'$ with delayed inversion)
  - As seen from the K-map, no loop covers both minterms $ABC$ and $AB'C$

```
<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>
```

Both 0 for 10ns after $B$ changes
(assume propagation delay 10ns for all gates)

Hazards in Combinational Circuits
Static 1-Hazard

Example (cont'd)

```
<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>
```

$D$ and $E$ both 0 momentarily

Timing Chart

Glitch in $F$
Hazard Detection and Removal

- **Hazard detection** procedure for 2-level AND-OR circuits
  1. Derive SOP expression of the circuit
  2. Plot each product term on K-map and loop it
  3. If any two adjacent 1’s are not covered by the same loop, a 1-hazard exists for the transition between these two 1’s
    - For an n-variable map, this transition occurs when one variable changes and the other n-1 variables are held constant (due to adjacent 1’s)

- **Hazard removal** for 2-level AND-OR circuits
  - Add a loop to the map covering such adjacent 1’s, and then add the corresponding gate to the circuit

Example (cont’d)

- Circuit with hazard removed
  - The new added product term AC keeps on 1 (for A=C=1) when B changes from 1 to 0
Hazards in Combinational Circuits

Static 0-Hazard

Example

- Letting A=0, B=1, D=0, and C from 0 to 1 results in a static 0-hazard \((Z = C \cdot C'\) with delayed inversion)

\[
\begin{array}{c}
\text{at 5ns, } 0 \rightarrow 1 \\
\text{at 10ns, } 0 \rightarrow 1 \\
\text{at 15ns, } 0 \rightarrow 1 \\
\text{at 18ns, } 1 \rightarrow 0 \\
\text{at 8ns, } 1 \rightarrow 0 \\
\end{array}
\]

(assume 3ns delay for inverters and 5ns for AND/OR gates)

Timing Diagram

Hazards in Combinational Circuits

Static 0-Hazard

Example (cont'd)

Timing Diagram
Hazards in Combinational Circuits

Static 0-Hazard

Example (cont'd)

- Eliminate the 0-hazards by adding 3 additional loops
  \[ Z = (A+C)(A'+D')(B'+C'+D)(C+D')(A+B'+D)(A'+B'+C') \]

<table>
<thead>
<tr>
<th>AB</th>
<th>CD</th>
<th>00</th>
<th>01</th>
<th>11</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>01</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Hazard-Free Circuit Design

- Procedure for the design of circuits free of static and dynamic hazards
  1. Find an SOP expression \( F_t \) for the output in which every pair of adjacent 1's is covered by a 1-term. (The sum of all prime implicants will always satisfy this condition.) A two-level AND-OR circuit based on this \( F_t \) will be free of 1-, 0-, and dynamic hazards
  - Alternatively can start with a POS expression in which every pair of adjacent 0's is covered by a 0-term, and follow the dual procedure to design a hazard-free two-level OR-AND circuit
  2. If a different form of the circuit is desired, manipulate \( F_t \) to the desired form by simple factoring, DeMorgan's laws, etc. Treat each \( x_i \) and \( x_i' \) as independent variables to prevent introduction of hazards
Hazards in Combinational Circuits

- Under what condition can static 0-hazard (1-hazard) appear in 2-level AND-OR (OR-AND) circuits?

Verification of Logic Circuits

- Verification may take 70% of an entire circuit design time!

  - Verification methodologies:
    - Formal verification
      - Mathematical proof of design correctness
    - Informal verification
      - Error identification by simulation (focus of textbook)

  - Verification objectives:
    - Functional verification
      - for logical correctness
    - Timing verification
      - for timing correctness

  - Testing
    - for quality control of fabricated ICs (focus of textbook)
    - simulation of faulty components in the circuit as an aid to finding tests for the circuit
Simulation and Testing of Logic Circuits

- Logic circuits can be verified by actually building them or by simulating them on computers
  - Simulation is easier, faster, and more economical
  - Computer simulation involves specifying a circuit, specifying its inputs, and observing its outputs

Simulation Procedure

- Simulation procedure for combinational circuits:
  1. Valuations are performed level by level (from inputs to outputs); changes are updated from gate inputs to gate outputs
    - 0, 1, X, Z four-valued simulation
      - X: unknown value (different from don't cares!)
      - Z: open circuit or high impedance (hi-Z)
  2. Step 1 is repeated until no more changes. The circuit is then in steady-state condition, and the outputs can be read
  3. Steps 1-2 are repeated every time a circuit input changes
Simulation and Testing of Logic Circuits

Four-Valued Logic Simulation

Probe (observation point)

<table>
<thead>
<tr>
<th>AND</th>
<th>0</th>
<th>1</th>
<th>X</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>X</td>
<td>0</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Z</td>
<td>0</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OR</th>
<th>0</th>
<th>1</th>
<th>X</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>1</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Z</td>
<td>X</td>
<td>1</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Simulation and Testing of Logic Circuits

Error Causes

Possible causes of wrong outputs for some set of input values, e.g.,

- In simulation
  - Incorrect design
  - Gates connected wrong
  - Wrong input signals to the circuit

- In fabricated circuit (built in lab)
  - Defective gates
  - Defective connecting wires
Example

- Locate the error for the circuit with function $F = AB(C'D+CD')+A'B'(C+D)$. Assume it, after built in lab, has an incorrect output equal to 1 when $A=B=C=D=1$.
  - By locating errors from the output to inputs, the inconsistency between the inputs and output of gate 3 is identified.
    - Inputs to gate 3 can be connected wrong, gate 3 is defective, or input connections to gate 3 can be defective.

![Observed circuit value diagram](image-url)