Lecture 11

- Timing diagrams
- Hazards

Timing diagrams (waveforms)

- Shows time-response of circuits
- Like a sideways truth table
- Example: $F = A + BC$

Timing diagrams

- Real gates have real delays
- Example: $A' \cdot A = 0$?

F = A + BC in 2-level logic

- Minimized product-of-sums
- Minimized sum-of-products
- Canonical product-of-sums
- Canonical sum-of-products

Timing diagram for F = A + BC

- Time waveforms for $F_1$ – $F_4$ are identical except for glitches

Hazards and glitches

- **glitch**: unwanted output
- A circuit with the potential for a glitch has a hazard.

- Glitches occur when different pathways have different delays
  - Causes circuit noise
  - Dangerous if logic makes a decision while output is unstable
Hazards and glitches

Solutions
- Design hazard-free circuits
- Difficult when logic is multilevel
- Wait until signals are stable

Types of hazards

- Static 1-hazard
  - Output should stay logic 1
  - Gate delays cause brief glitch to logic 0

- Static 0-hazard
  - Output should stay logic 0
  - Gate delays cause brief glitch to logic 1

- Dynamic hazards
  - Output should toggle cleanly
  - Gate delays cause multiple transitions

Static hazards

- Often occurs when a literal and its complement momentarily assume the same value
  - Through different paths with different delays
  - Causes an (ideally) static output to \textit{glitch}

Timing diagram for $F = A + BC$

$F = A + BC$ in 2-level logic

A multiplexer

$F = A + BC$ in 2-level logic

canonical product-of-sums
Timing diagram for \( F = A + BC \)

\[
\begin{array}{cccc}
A & B & C & F \\
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 \\
0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 \\
1 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 \\
1 & 1 & 0 & 0 \\
1 & 1 & 1 & 1 \\
\end{array}
\]

\( F = A + BC \) in 2-level logic

Dynamic hazards
- Often occurs when a literal assumes multiple values
  - Through different paths with different delays
  - Causes an output to toggle multiple times

Eliminating static hazards
- Key idea: Glitches happen when a changing input spans separate K-map encirclements
  - Example: 1101 to 0101 change can cause a static-1 glitch

\begin{align*}
F &= AC' + A'D \\
\end{align*}
Eliminating static hazards

- Solution: Add redundant K-map encirclements
  - Ensure that all single-bit changes are covered by same block
  - First eliminate static-1 hazards: Use SOP form
  - If need to eliminate static-0 hazards, use POS form
- Technique only works for 2-level logic

Eliminating static hazards

\[ F = AC' + A'D' + CD \]

Summary of hazards

- We can eliminate static hazards in 2-level logic for **single-bit changes**
  - Eliminating static hazards also eliminates dynamic hazards
- Hazards are a difficult problem
  - Multiple-bit changes in 2-level logic are hard
  - Static hazards in multilevel logic are harder
  - Dynamic hazards in multilevel logic are harder yet