10-Combinational Circuit Design

Text: Unit 8

ECEGR/ISSC 201
Digital Operations and Computations
Winter 2011

Overview

- Gates with Limited Fan-in
- Delays and Timing
- Other Hazards

Limited Fan-in

- Practical logic gates are limited by the number of inputs they can handle (fan-in)
- Similar constraints on the number of gates the output can be connected to (fan-out)
- Solution to limited fan-in: factor into a multi-level expression
- Solution to limited fan-out: use buffers (discussed in the next lecture)

Limited Fan-in

- Consider F(A, B, C, D) = Σm(0, 3, 4, 5, 8, 9, 10, 14, 15)
- Design using NOR gates

CD \ AB	00	01	11	10
00	1	1	0	1
01	0	1	0	1
11	1	0	1	0
10	0	0	1	1

Limited Fan-in

- Recall NOR-NOR requires starting with PoS form
- F' = A'B'C'D + AB'CD + ABC' + A'BC + A'CD'

CD \ AB	00	01	11	10
00	1	1	0	1
01	0	1	0	1
11	1	0	1	0
10	0	0	1	1

Limited Fan-in

F' = A'B'C'D + AB'CD + ABC' + A'BC + A'CD'

F =(A'B'C'D + AB'CD + ABC' + A'BC + A'CD')'

F=(A+B+C+D')(A'+B+C'+D')(A'+B'+C)(A+B'+C')(A+C'+D)

Next F = {(F)'}'

F={(A+B+C+D')'+(A'+B+C'+D')'+(A'+B'+C)'+(A+B'+C')'+(A+C'+D)'}'

Limited Fan-in

- F={(A+B+C+D')'+(A'+B+C'+D')'+(A'+B'+C)'+(A+B'+C')'+(A+C'+D)'}'
- This requires:
  - Three, 3-input NOR
  - Two, 4-input NOR
  - One, 5-input NOR
- What if we are limited to 3-input NORs only?

Limited Fan-in

- Start with:

F' = A'B'C'D + AB'CD + ABC' + A'BC + A'CD'

Factor to:

F' = B'D(A'C' + AC) + A'C(B+D') + ABC'

- Convert this to NOR-NOR form

Limited Fan-in

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]

Next F = {(F)'}'

Limited Fan-in

F = [B+D' +(A+C)(A'+C')][A+C'+B'D][A'+B'+C]

Next F = {(F)'}'

F = {([B+D' +(A+C)(A'+C')][A+C'+B'D][A'+B'+C])'}'

F= {[B+D'+(A+C)(A'+C')]'+[A+C'+B'D]'+[A'+B'+C]'}'

Using (A+C)(A'+C') = {[(A+C)(A'+C')]'}' and

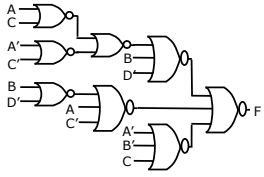
B'D = [(B'D)']' yields:

F= {[B+D'+((A+C)'+(A'+C')')']'+[A+C'+(B+D')']'+[A'+B'+C]'}'

Limited Fan-in

Build the circuit of

F= {[B+D'+((A+C)'+(A'+C')')']'+[A+C'+(B +D')']'+[A'+B'+C]'}'

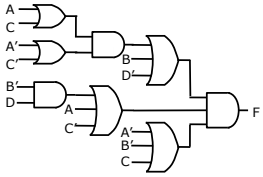


Limited Fan-in

F' = B'D(A'C' + AC) + A'C(B+D') + ABC'

Recall that you can build:

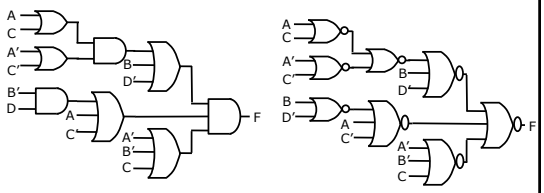
F = [B+D' +(A+C)(A'+C')][A+C'+B'D][A'+B'+C]



Limited Fan-in

F = [B+D' +(A+C)(A'+C')][A+C'+B'D][A'+B'+C]

And replace the gates with NORs, and invert the single literal inputs to levels 1, 3, 5 etc (see text page 188)



Limited Fan-in

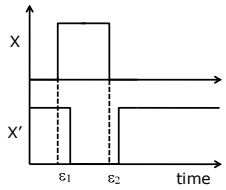
- Design procedure for multi-output circuits
  - Minimize individually (common terms are usually lost when factoring)
  - Factor individually as needed to meet fan-in requirements

Gate Delays

- Output changes do not occur instantaneously
- Propagation delay, ε
- On the order of 1 nanosecond
- Propagation delay is not symmetric
  - 0 to 1 ε_1
  - 1 to 0 ε_2
- In multi-level circuits, the delay can be important

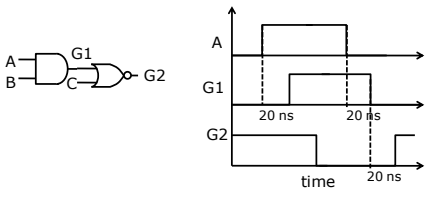
Gate Delays

- Example of an inverter propagation delay



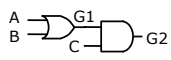
Example

- Assume the delays are 20 ns for each gate
- Let B = 1 and C =0, A changes as shown



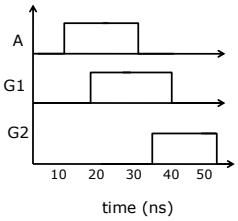
Example

- Draw the timing diagram if the delay for gate 1 is 10 ns and gate 2 is 15 ns
- Assume
  - A is low from 0 to 10ns and high from 10 ns to 30 ns
  - B = 0, and C = 1



Example

- Draw the timing diagram if the delay for gate 1 is 10 ns and gate 2 is 15 ns

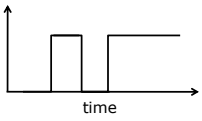


Hazards in Combinational Logic

- Transients can occur if propagation delays are different for the various IC
- Static 1-hazard: when the output should be a 1 but it momentarily goes to 0 in response to an input change
- Static 0-hazard: when the output should be a 0 but it momentarily goes to 1 in response to an input change

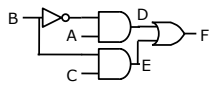
Hazards in Combinational Logic

- Dynamic hazard: when the output is supposed to change from 1 to 0 or 0 to 1 and the change occurs three or more times



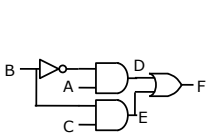
Hazards in Combinational Logic

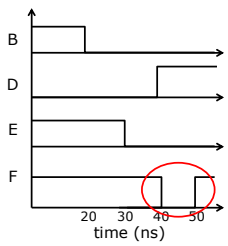
- Consider: F = AB' +BC
- Assume that
  - A=1
  - C=1
- Then: F = B' + B = 1
- F should remain at 1 when B changes from 0 to 1 or 1 to 0



Hazards in Combinational Logic

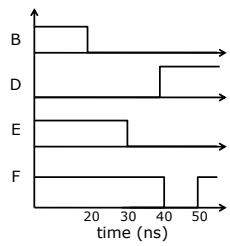
- Timing diagram:





Hazards in Combinational Logic

- Is this a static 1 or static 0 hazard?
- Static 1 hazard



Hazards in Combinational Logic

- We can identify potential hazards by examining K-maps
- No single loop covers ABC and AB'C
- If A=C=1 and B changes, both can momentarily go to 0

BC \ A	0	1
00	0	1
01	0	1
11	1	1
10	0	0

Hazards in Combinational Logic

- Procedure for examining K-maps for hazards
  - Write down SoP form of expression
  - Plot each term on the K-map and loop as necessary
  - If any two adjacent 1s are not covered by the same loop, a static 1 hazard exists for the transition between the two 1s
  - For an n-variable k-map, the transition occurs when one variable changes and the other n-1 variables are held constant

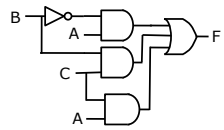
Hazards in Combinational Logic

- To remove the hazard

BC \ A	0	1
00	0	1
01	0	1
11	1	1
10	0	0

Hazards in Combinational Logic

- Circuit with the hazard removed



Hazards in Combinational Logic

- Static 0-hazards occur analogously with 0s on the K-map
- Procedure for the removal of the hazard is analogous to removal of static 1-hazards

Hazards in Combinational Logic

- Design of circuits without static hazards:
  1. Find a SoP expression of F in which every pair of adjacent 1s is covered by a 1-term
     - This can be done by summing all the prime implicants
     - A two-level AND-OR circuit will be free of 1- and 0-static hazards and dynamic hazards
  2. If a different form of F is needed, algebraically manipulate F, but treat literals and their complements as independent variables

Example

- Derive an expression for F that eliminates static 0-hazards

CD \ AB	00	01	11	10
00	0	0	1	1
01	0	0	0	0
11	1	1	0	0
10	1	0	0	1

Example

- Derive an expression for F that eliminates static 0-hazards
- F = (A+C)(A'+D')(B'+C'+D)(C+D')(A+B'+D)(A'+B'+C')