THE UNIVERSITY OF TEXAS AT DALLAS

CS4341 Digital Logic & Computer Design

Lecture Notes 11

Omar Hamdy
Assistant Professor
Department of Computer Science

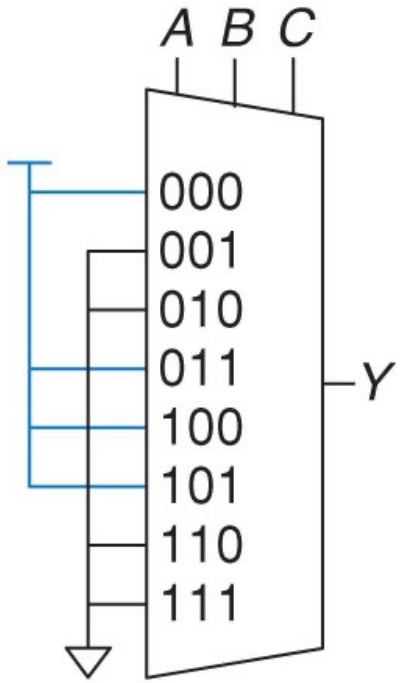
Example: Design Using MUX

- Design the Boolean function F = AB’ + B’C’ + A’BC using 8x1 mux

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

$$Y = A\overline{B} + \overline{B}\,\overline{C} + \overline{A}BC$$

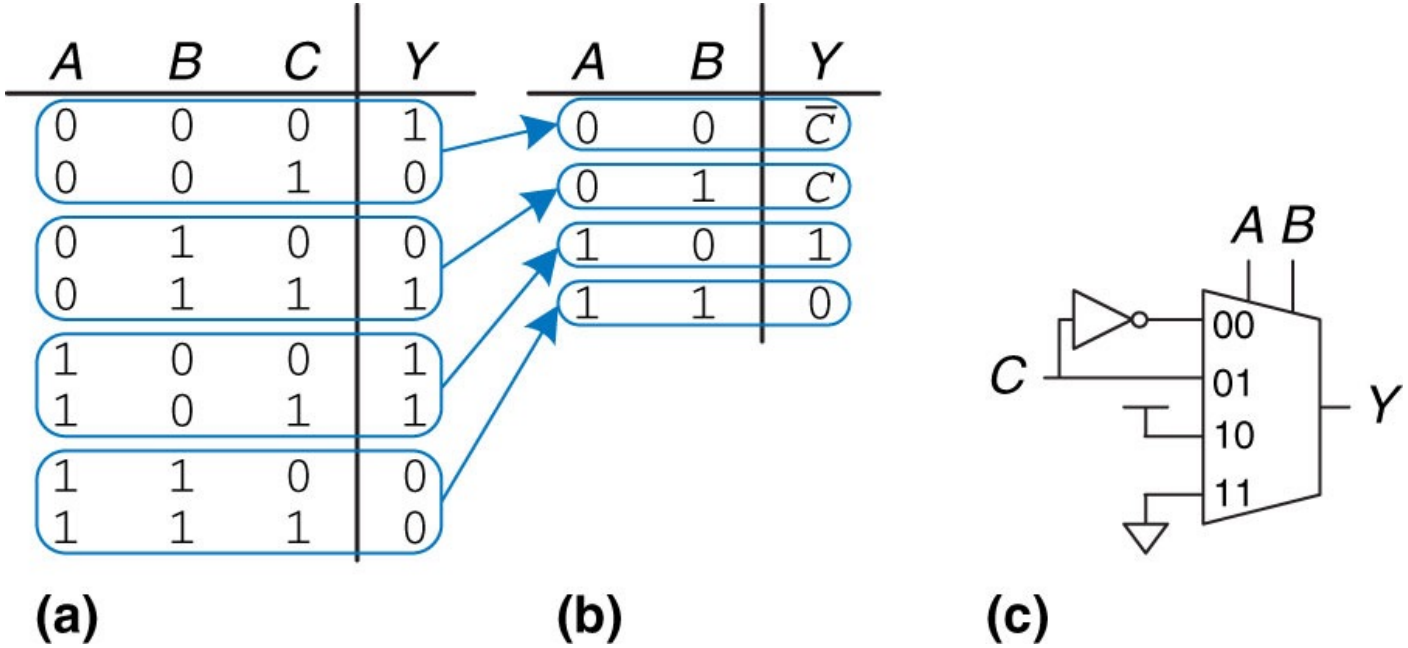
(a)



(b)

Example: Design Using MUX

- Design the Boolean function F = AB’ + B’C’ + A’BC using 4x1 mux



A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

(a)

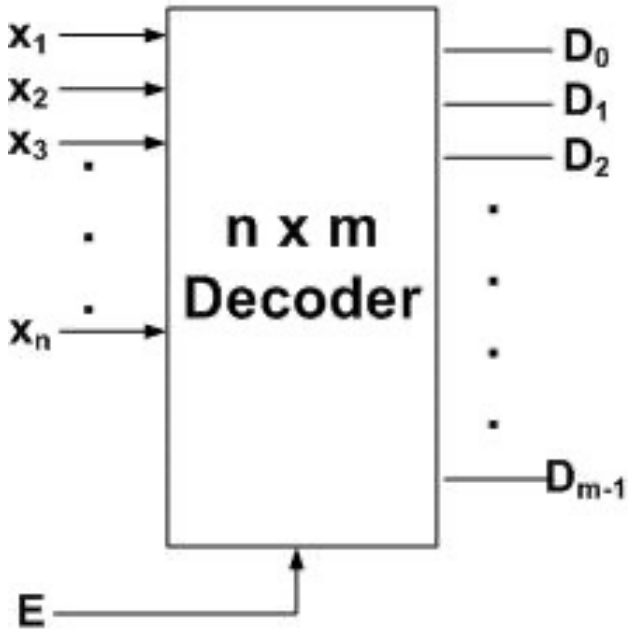
A	B	Y
0	0	$\overline{C}$
0	1	C
1	0	1
1	1	0

(b)

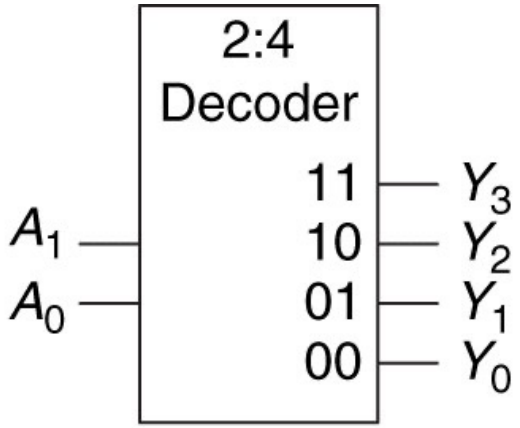
(c)

Decoders

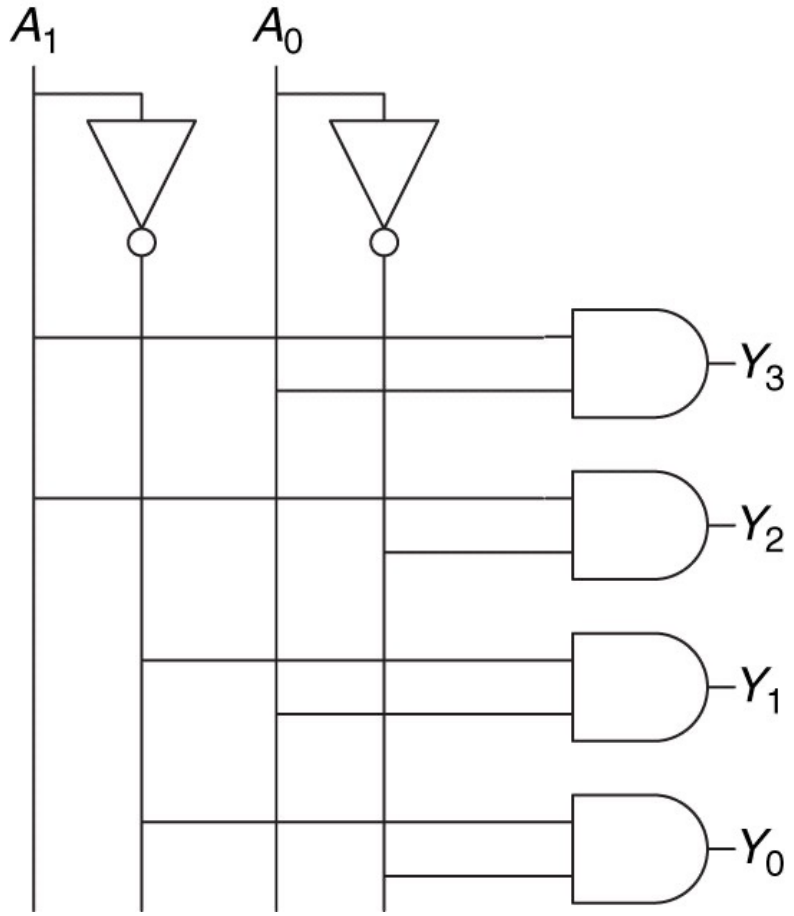
- An n x m decoder is a combinational circuit that coverts binary information from n input lines to m output lines, where $m \leq 2^n$
- Output values of decoders are mutually exclusive: exactly one output is 1 for any given input combination **(address).**



2:4 Decoder



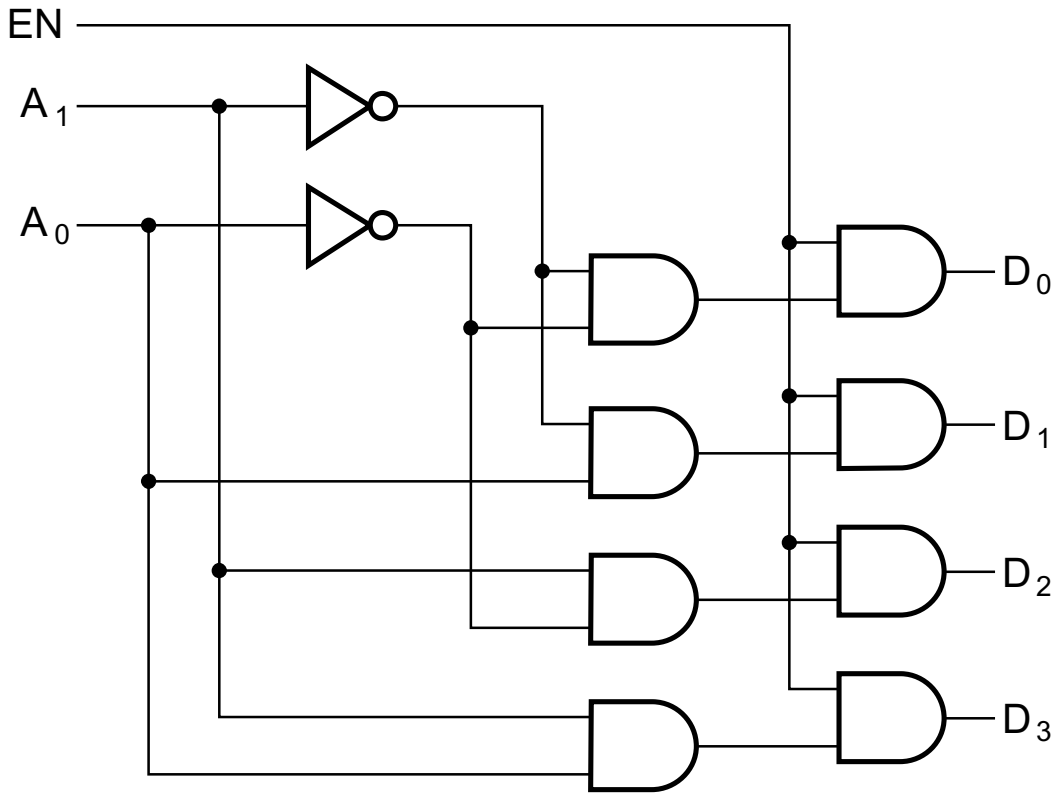
A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0



Decoders with Active-High Enable *E*

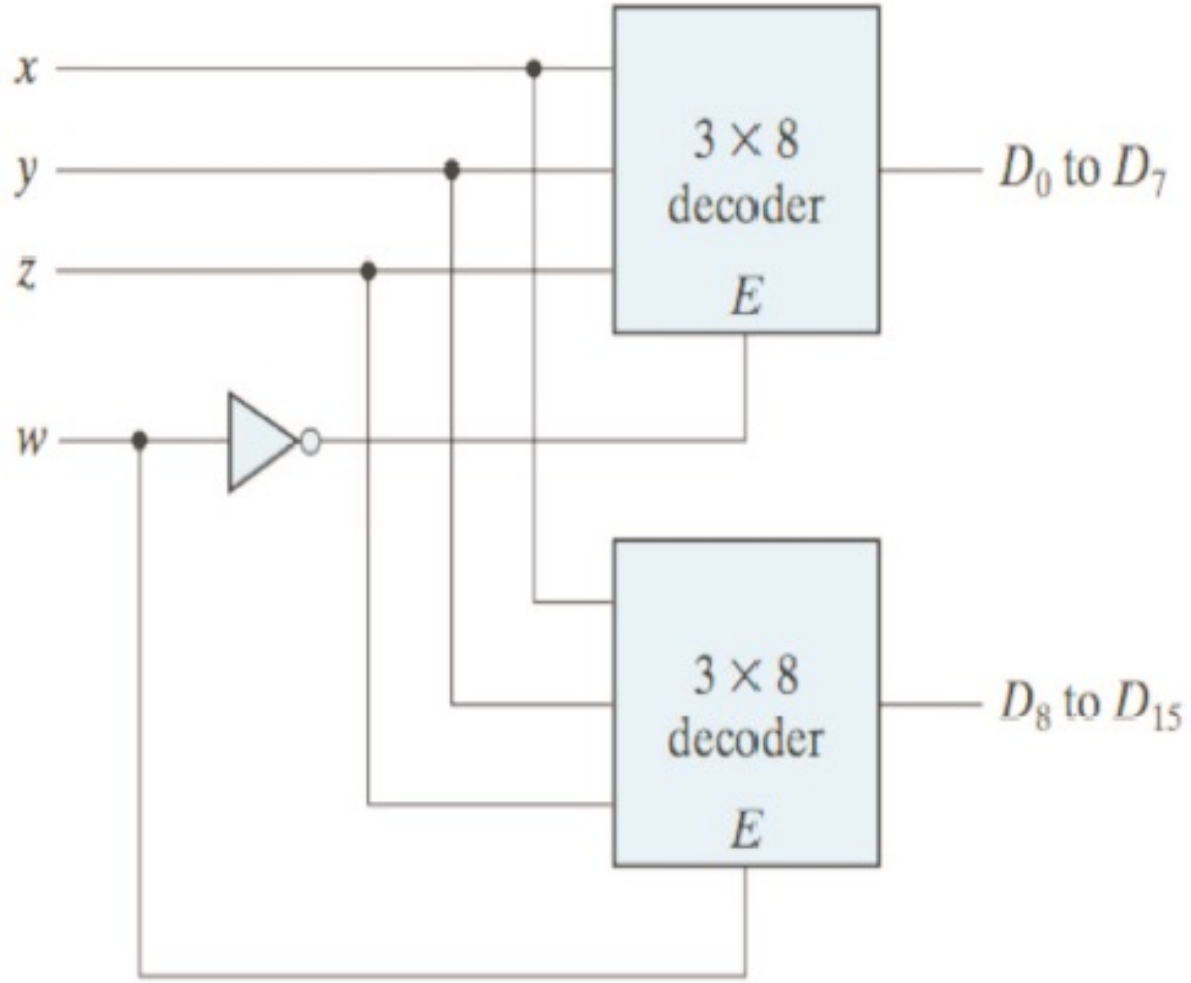
EN	A_1	A_0	D_0	D_1	D_2	D_3
0	X	X	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

(a)



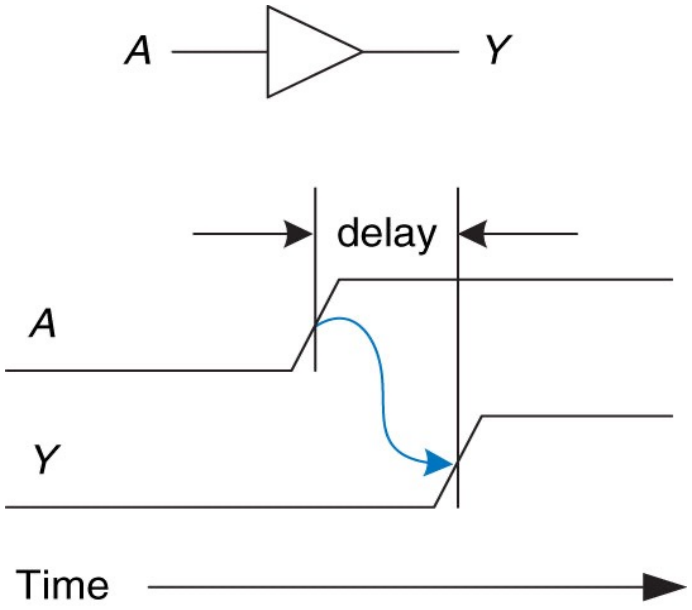
(b)

Hierarchical Decoder Design



Timing

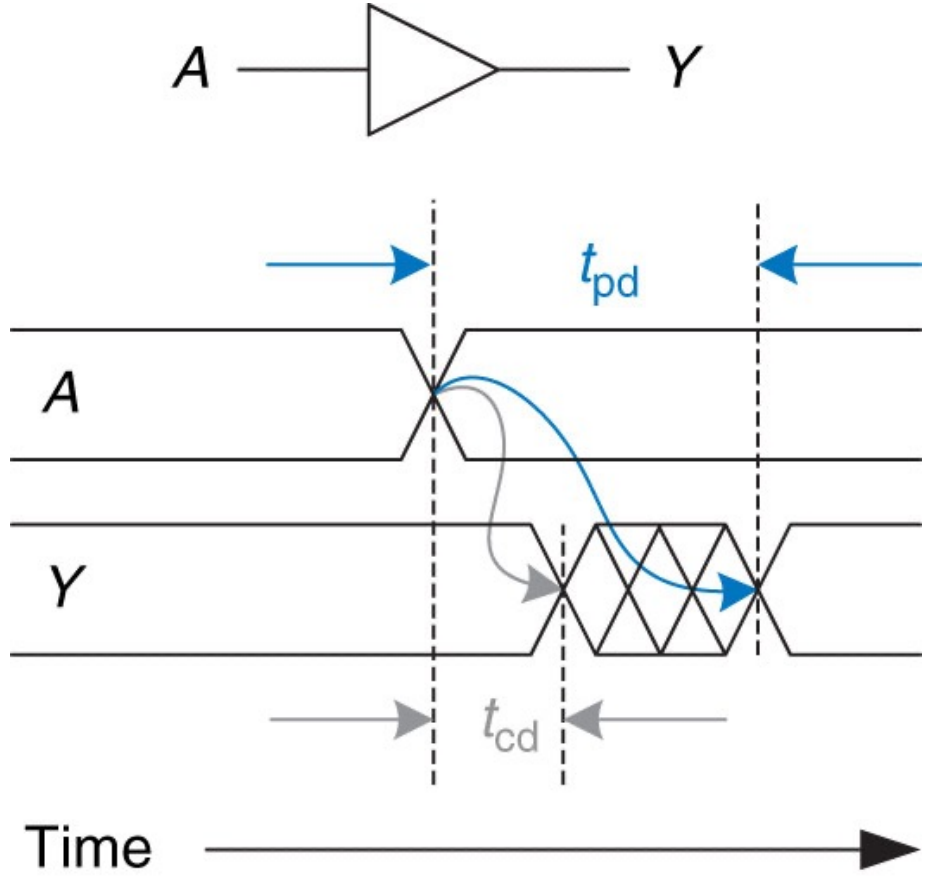
- Until now, we assumed there is a 0-time for input values to propagate through the combinational circuit to produce the output.
- The reality is, time is one of the key design factors as output takes time to change as the input signals pass through the circuit.



- A timing diagram shows the transient response of the circuit when input changes.
- Transition from Low to High is called rising edge
- Transition from High to Low is called falling edge
- A 50% point is the point when a signal is half way between High and Low
- Delay is measured from the 50% point of the input signal to the 50% point of the output signal

Propagation and Contamination Delay

- Combinational logic is characterized by its propagation delay and contamination delay.
- The propagation delay, t_{pd}, is the maximum time from when an input changes until the output or outputs reach their final value.
- The contamination delay, t_{cd}, is the minimum time from when an input changes until any output starts to change its value

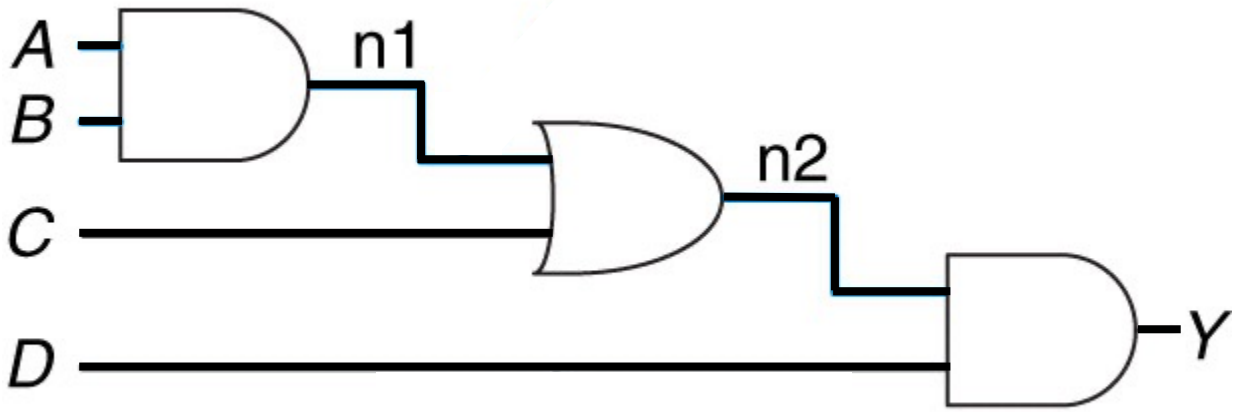


Delay Calculation

- At high-level, propagation and contamination delays are determined by the path a signal takes from input to output.
- Two paths are usually calculated:
  - The critical path, which is the longest time-delay inputs will take to change the output (overall circuit maximum speed)
  - The short path, the shortest time-delay inputs that can start changing the output (in some cases could be the best-case scenario).
  - The propagation delay of a combinational circuit is the sum of the propagation delays through each element on the critical path
  - The contamination delay is the sum of the contamination delays through each element on the short path

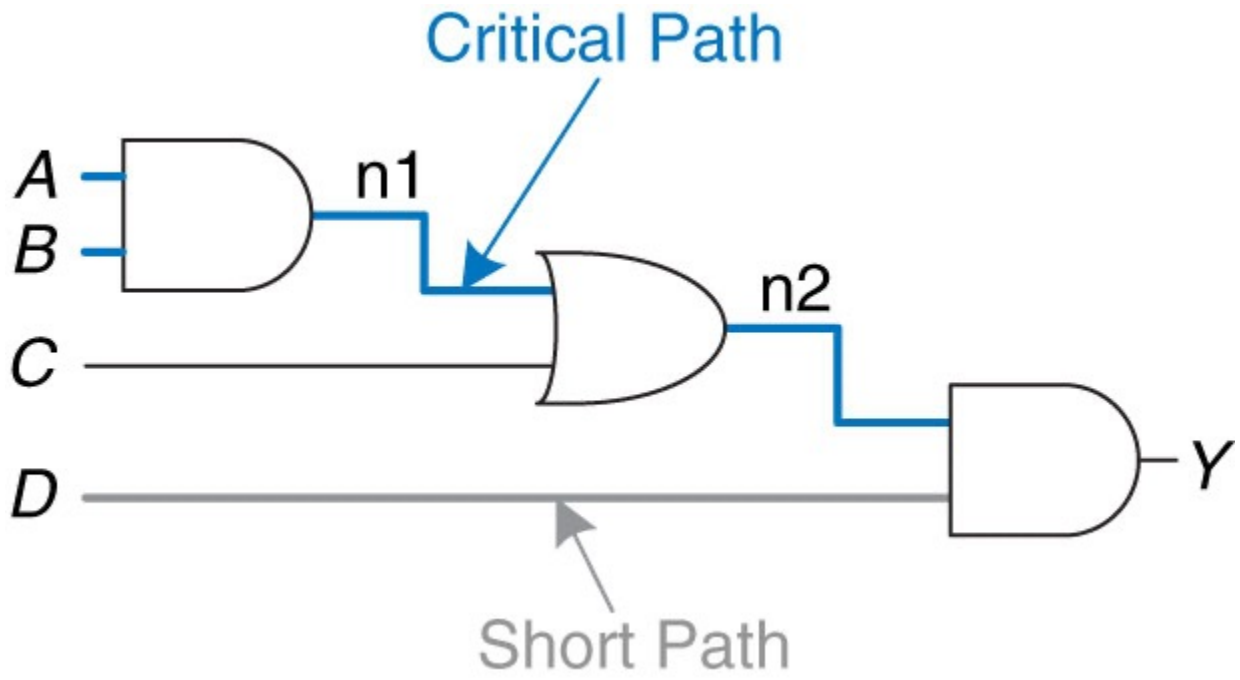
Delay Calculation Example

- Find t_{pd} and t_{cd} for the following circuit diagram

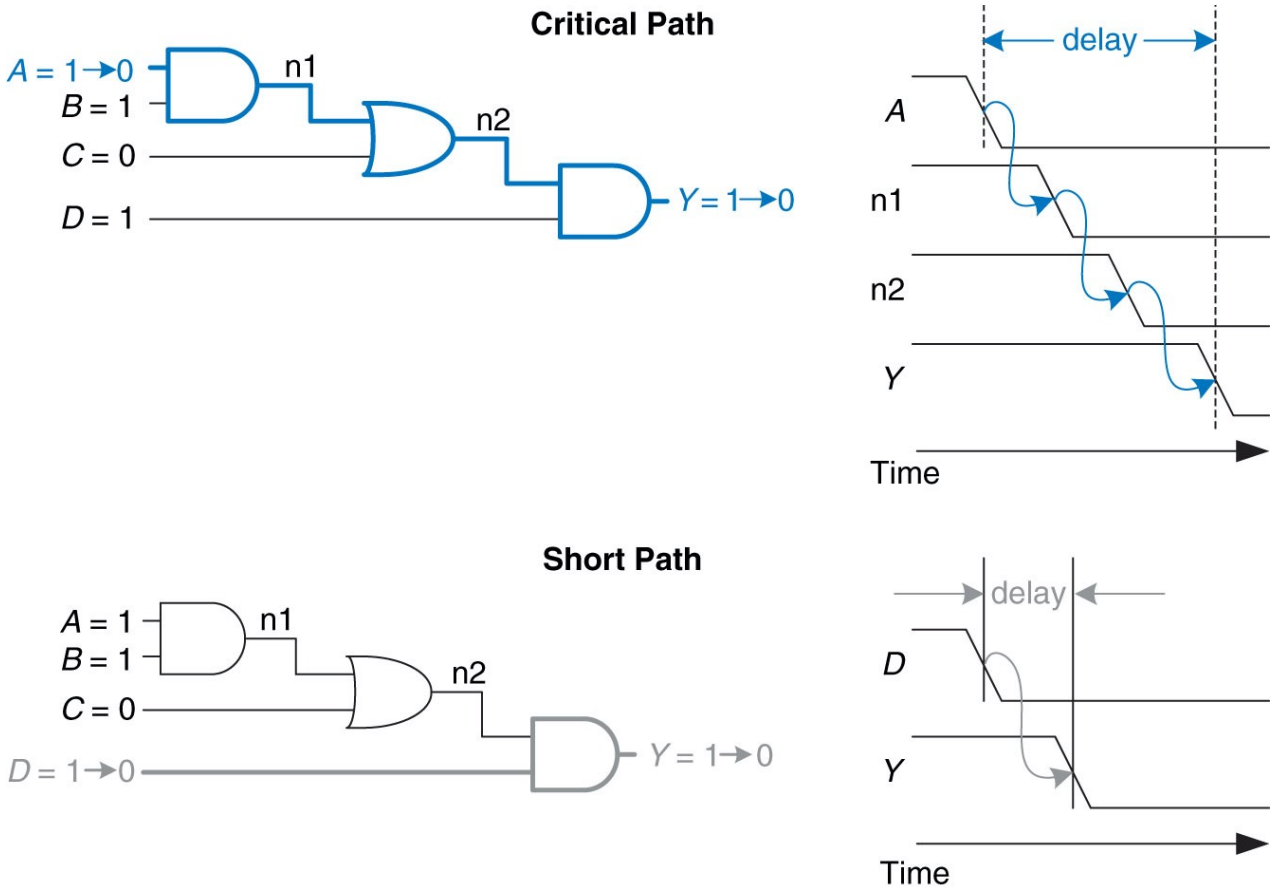


Delay Calculation Example

- First, we determine critical and short paths



Delay Calculation Example



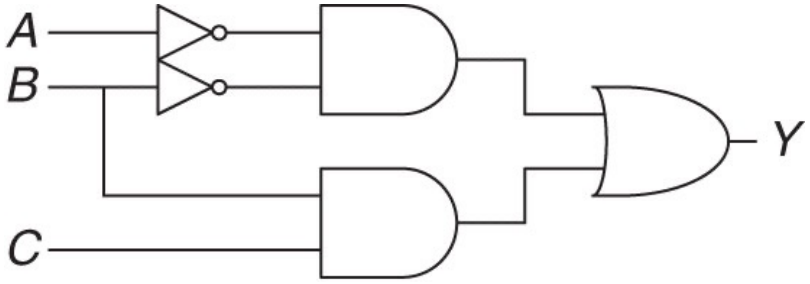
- $t_{pd} = 2 \times t_{pd\_AND} + t_{pd\_OR}$
- $t_{cd} = t_{cd\_AND}$

Time Glitches

- Sometimes the circuit design will cause the output to do multiple transitions before it becomes steady
- Usually that is not an issue as long as we wait until the propagation delay elapse.

Glitch Example

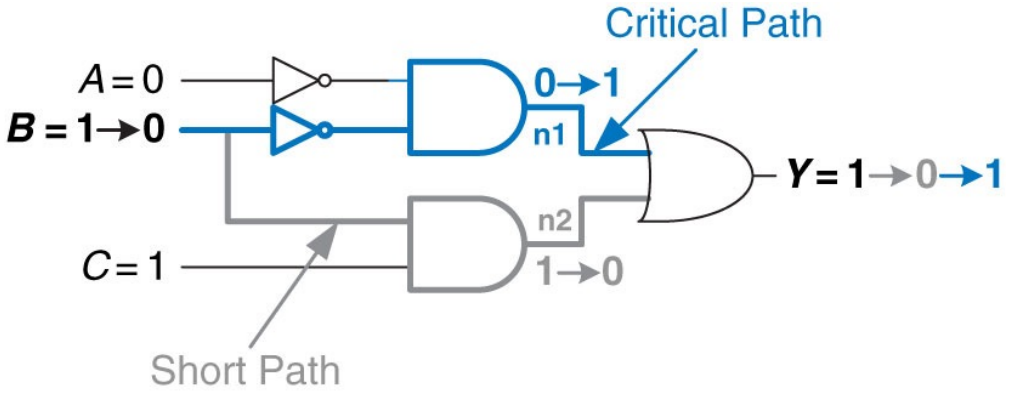
- Analyze the following circuit and determine the output transitioning sequence

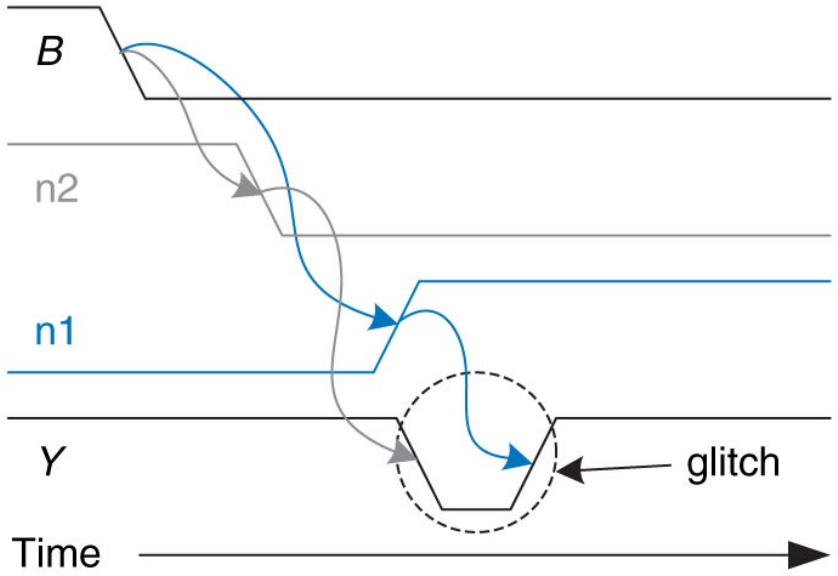


Y	AB = 00	AB = 01	AB = 11	AB = 10
C = 0	1	0	0	0
C = 1	1	1	1	0

$$Y = \overline{A}\,\overline{B} + BC$$

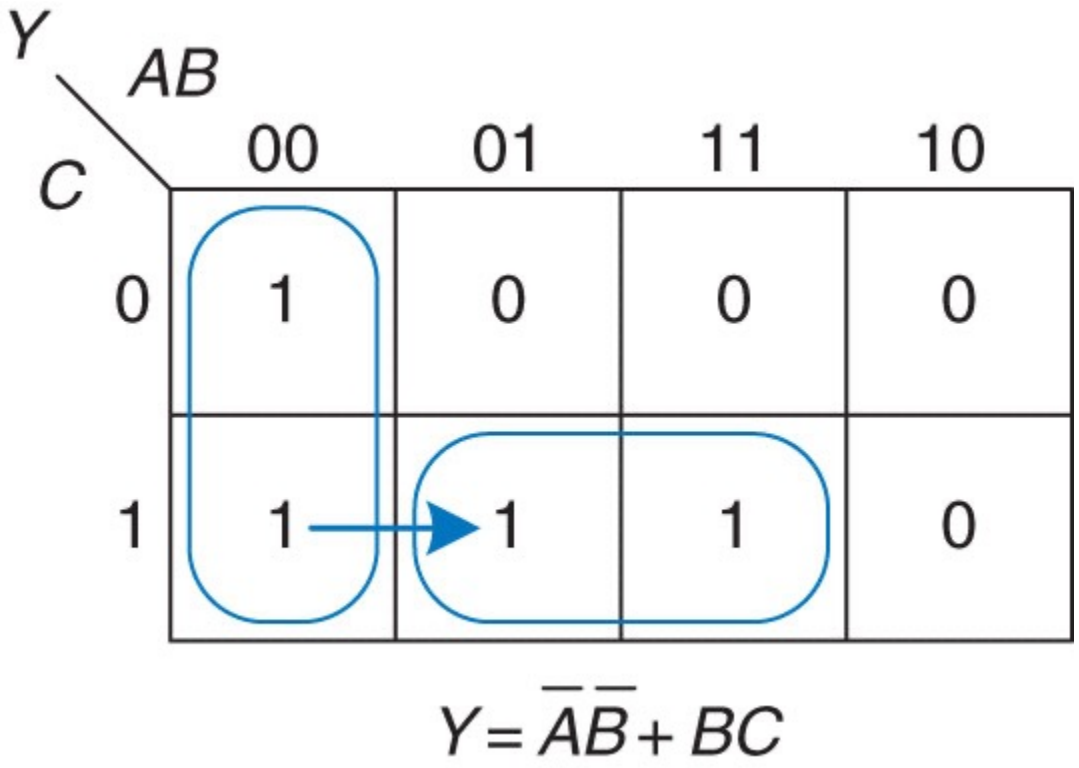
Glitch Example





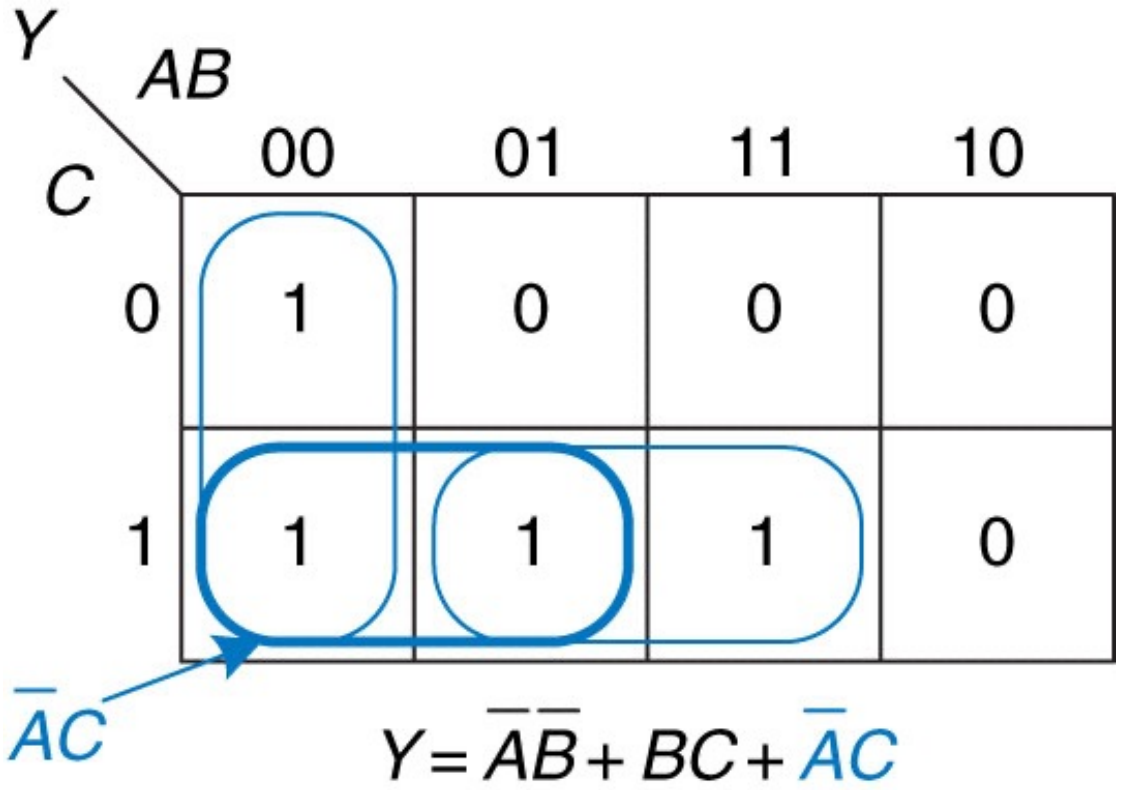
Avoiding Glitches

- From K-Map, a transition from one prime implicant to another could be a potential glitch

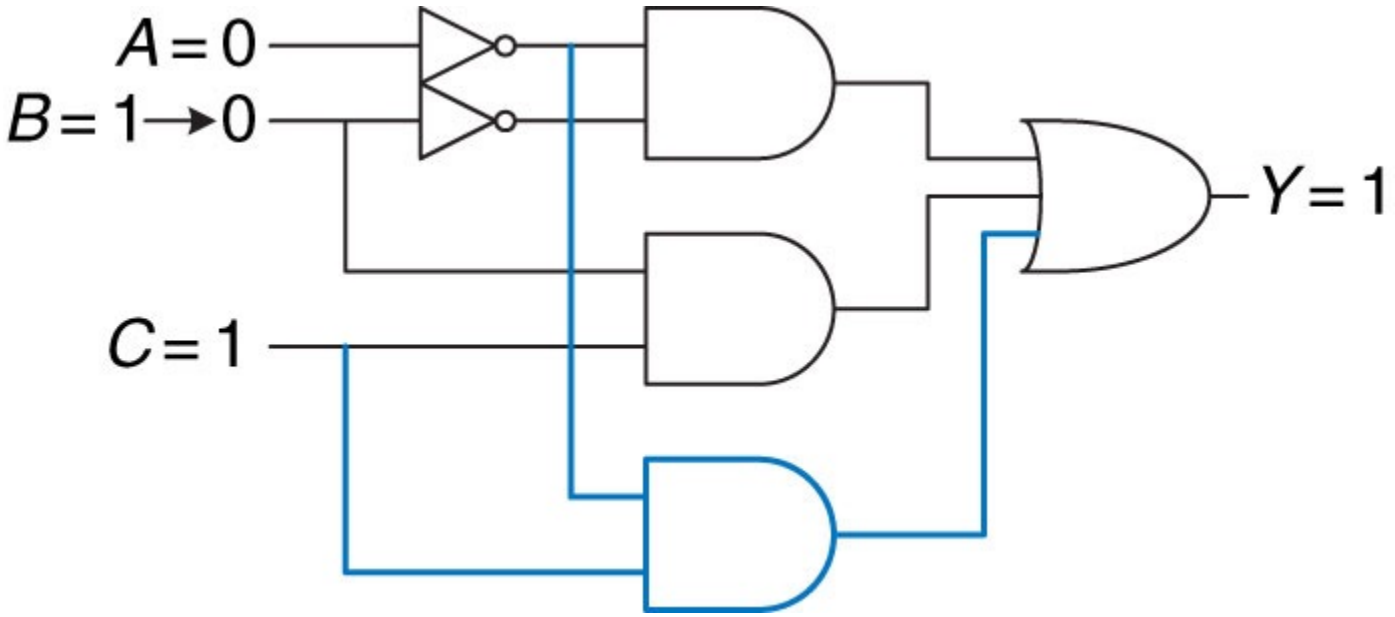


Avoiding Glitches

- Glitches can be avoided by overlapping prime implicants with additional redundant prime implicants
- This means additional gates added to the circuit.

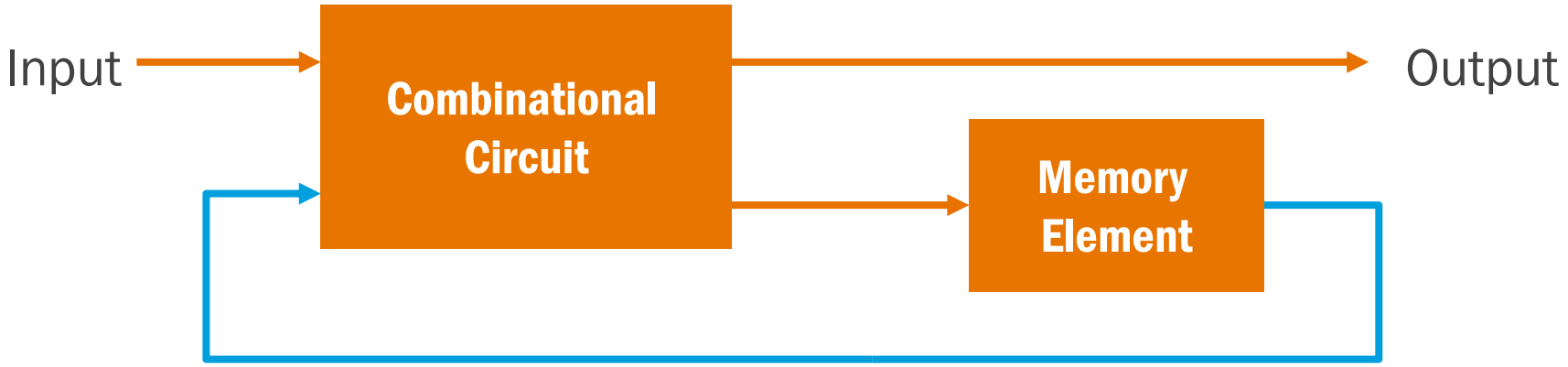


Glitch-Free Circuit



Sequential Circuits

- Sequential circuits are digital circuits that the output depends on:
  - current input (combinational circuit), and
  - system state: a memory device to preserve the output value



Example

- What is the key difference between these two locks?

Combinational: output depends only on input

Sequential: output depends on new and previous inputs

Sequential Circuit Types

- **Sequential circuits are differentiated by times at which:**
  - storage elements observe their inputs, and
  - storage elements change their state
- **Synchronous sequential circuits:**
  - Behavior is defined from knowledge of inputs at discrete instances of time
  - Storage elements observe inputs and can change state only in relation to a timing signal (clock pulses from a clock)
- **Asynchronous sequential circuits:**
  - Behavior is defined from knowledge of inputs an any instant of time
  - Storage elements observe inputs and can change state at any instant of time

To Do List

- Review lecture notes