XMUT 202
Digital Electronics

Felix Yan

School of Engineering and Computer Science
Victoria University of Wellington

Victoria
UNIVERSITY OF WELLINGTON
Te Whare Wānanga
o te Ūpoko o te Ika a Māui

CAPITAL CITY UNIVERSITY

Test yourself:

Use a K-map to simplify:

$$y = \overline{C}(\overline{A}\overline{B}\overline{D} + D) + A\overline{B}C + \overline{D}$$

$$y = \overline{A}\overline{B}\overline{C}\overline{D} + \overline{C}D + A\overline{B}C + \overline{D}$$

	$\overline{C}\overline{D}$	$\overline{C}D$	CD	$C\overline{D}$
$\overline{A}\overline{B}$	1	1	0	1
$\overline{A}B$	1	1	0	1
AB	1	1	0	1
$A\overline{B}$	1	1	1	1

$$y = A\overline{B} + \overline{C} + \overline{D}$$

Exclusive OR and Exclusive NOR Circuits

- The exclusive OR, abbreviated XOR produces a HIGH output whenever the two inputs are at opposite levels.
- The exclusive NOR, abbreviated XNOR produces a HIGH output whenever the two inputs are at the same level.
- XOR and XNOR outputs are opposite.

FIGURE 4-20 (a) Exclusive-OR circuit and truth table; (b) traditional XOR gate symbol; (c) IEEE/ANSI symbol for XOR gate.

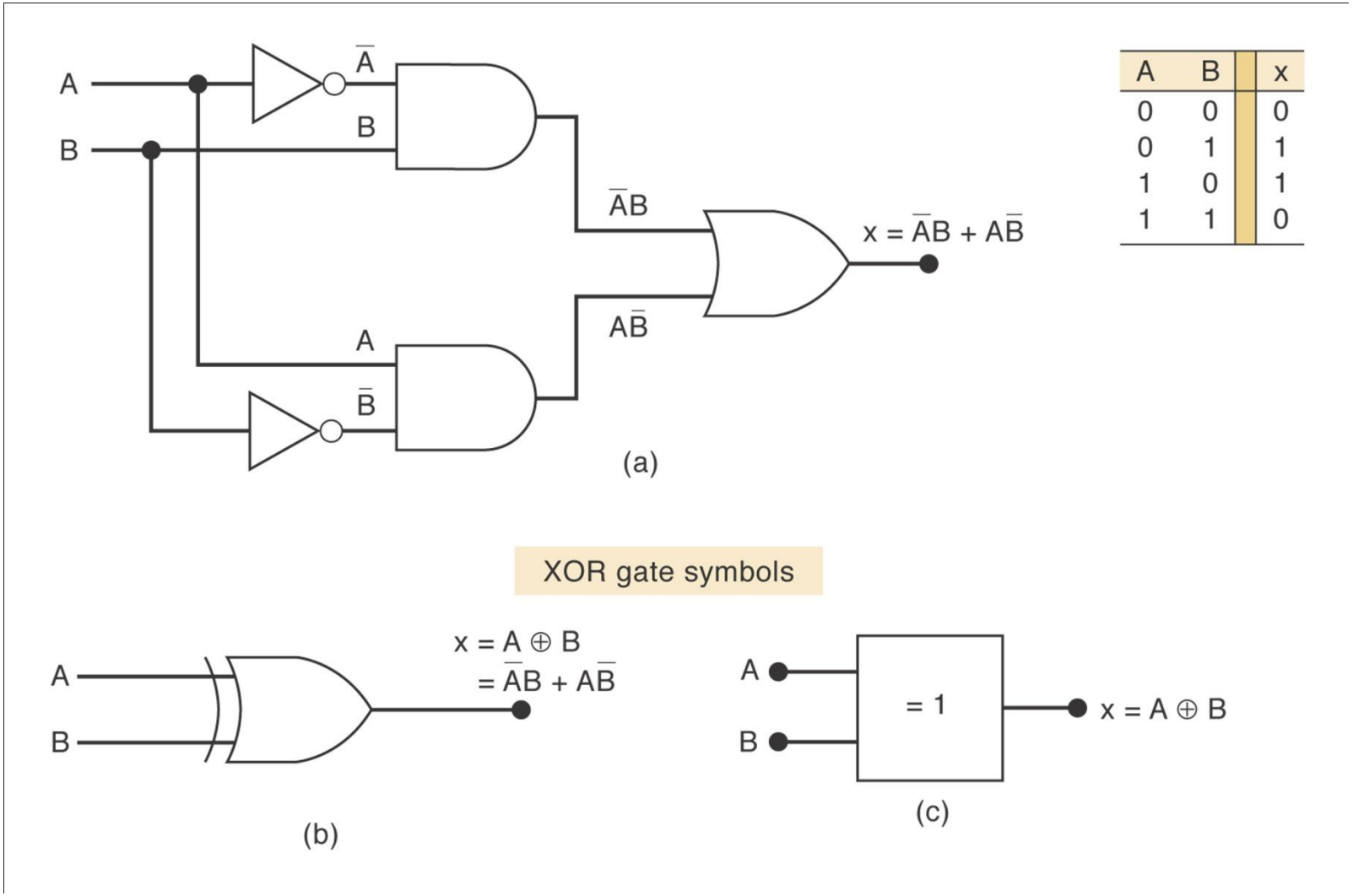
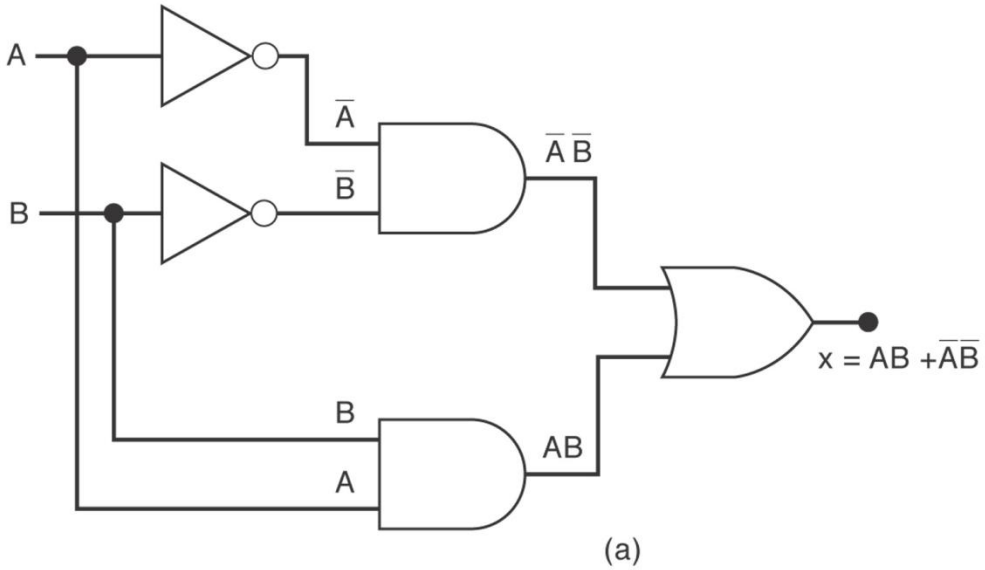


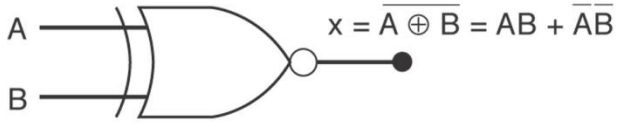
FIGURE 4-21 (a) Exclusive-NOR circuit; (b) traditional symbol for XNOR gate; (c) IEEE/ANSI symbol.



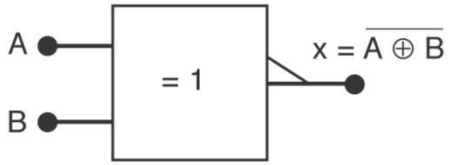
A	B	x
0	0	1
0	1	0
1	0	0
1	1	1

(a)

XNOR gate symbols

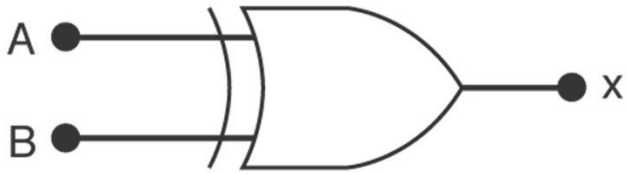


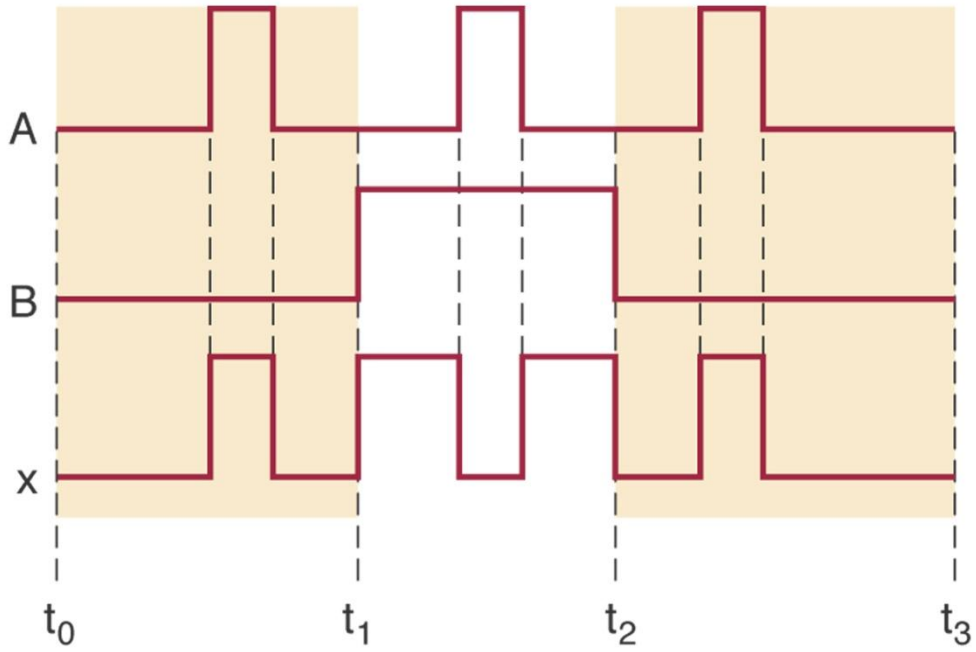
(b)



(c)

Determine the O/P waveform of the circuit below:

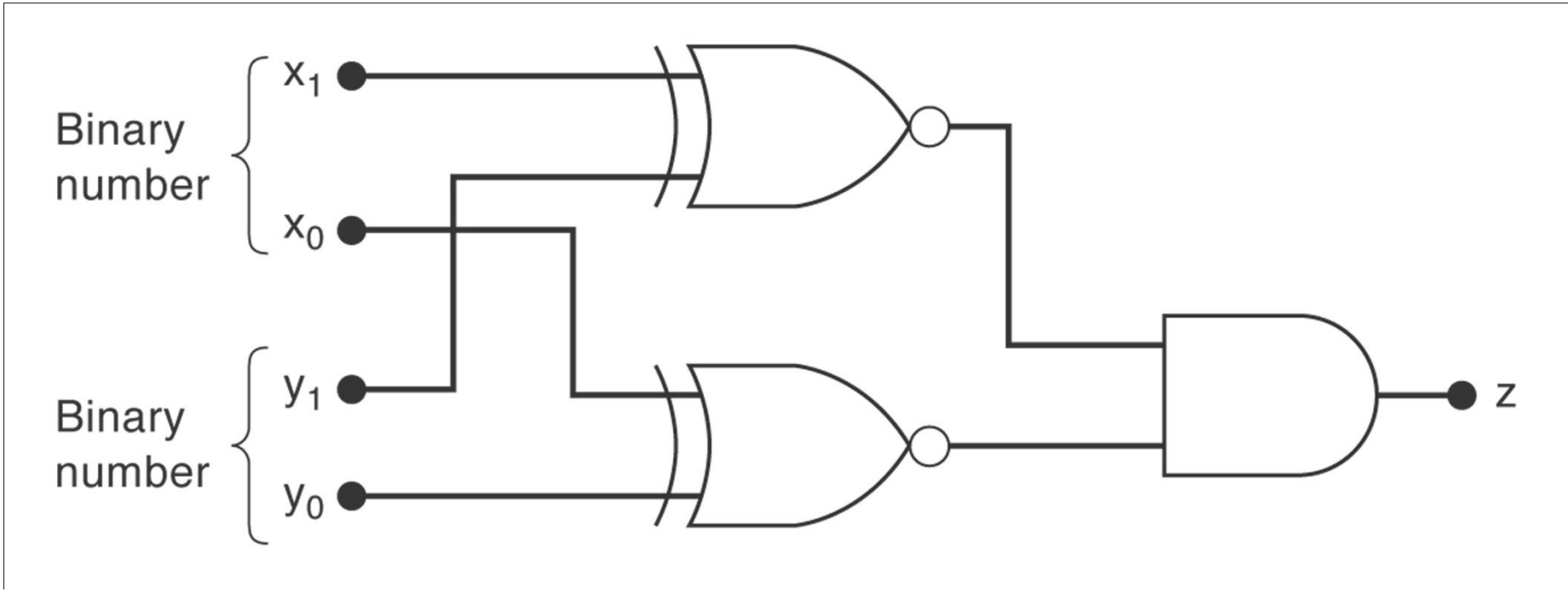




O/P Hi when I/P at different levels

Design a circuit so that the O/P will only be Hl when the combination of two sets of two bit binary numbers are equal.

x_1	x_0	y_1	y_0	z (Output)
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1


Binary number
x_1
x_0
Binary number
y_1
y_0
z

Parity Generator and Checker

- Parity bit: extra bit included with data to make the number of 1's even (for even parity) or odd (for odd parity)
- It is used to detect error in transmission
- Example: if we use an even parity system:
  - Data: 1 1 1 0 – we add a parity bit 1
  - Data: 1 1 0 0 – we add parity bit 0
  - Data: 0 0 0 0 – we add a parity bit 0

- A Parity Checker will return a TRUE error bit if the number of 1's is odd (for even parity) and if the number of 1's is even (for odd parity)

Parity Generator

$$A \oplus B = \overline{A}B + A\overline{B} \qquad \overline{A \oplus B} = \overline{A}\,\overline{B} + AB$$

- How to construct an even parity generator (3 bits input)?
- Truth table:

A	B	C	P
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Parity Generator

$$A \oplus B = \overline{A}B + A\overline{B} \qquad \overline{A \oplus B} = \overline{A}\,\overline{B} + AB$$

- How to construct an even parity generator (3 bits input)?
- Truth table:

A	B	C	P
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

$$P = \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\,\overline{C} + ABC$$

Parity Generator

$$A \oplus B = \overline{A}B + A\overline{B} \qquad \overline{A \oplus B} = \overline{A}\overline{B} + AB$$

- How to construct an even parity generator (3 bits input)?
- Truth table:

A	B	C	P
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

$$\begin{aligned} P &= \overline{A}\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\overline{C} + ABC \\ &= \overline{A}(\overline{B}C + B\overline{C}) + A(\overline{B}\overline{C} + BC) \\ &= \overline{A}(B \oplus C) + A(\overline{B \oplus C}) \\ &= \overline{A}X + A\overline{X} \\ &= A \oplus B \oplus C \end{aligned}$$

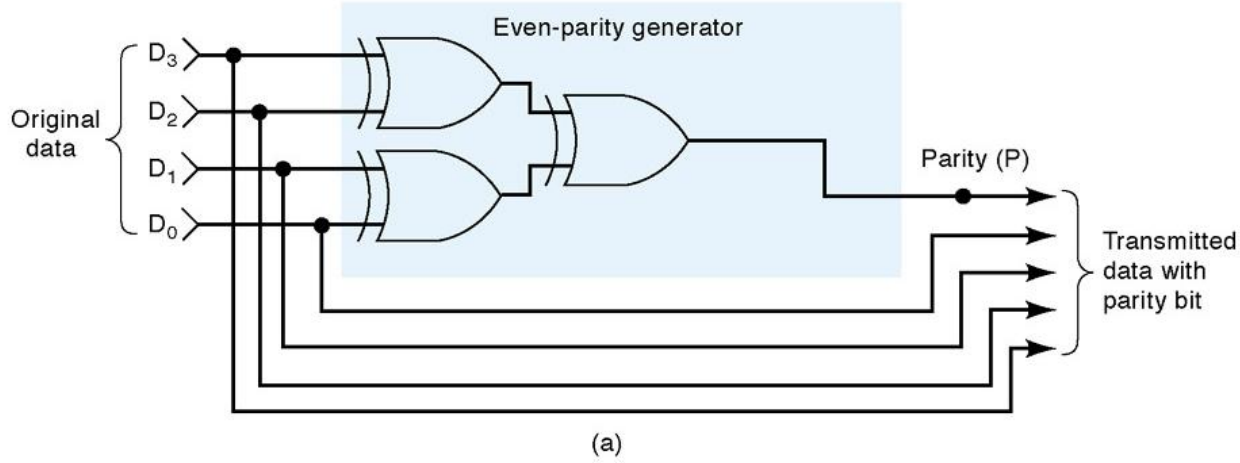
Parity Checker

- Homework exercise:
  - Design an even parity checker (3 data bits) using a truth table
  - Express it using XOR or XNOR gates

A	B	C	P	E

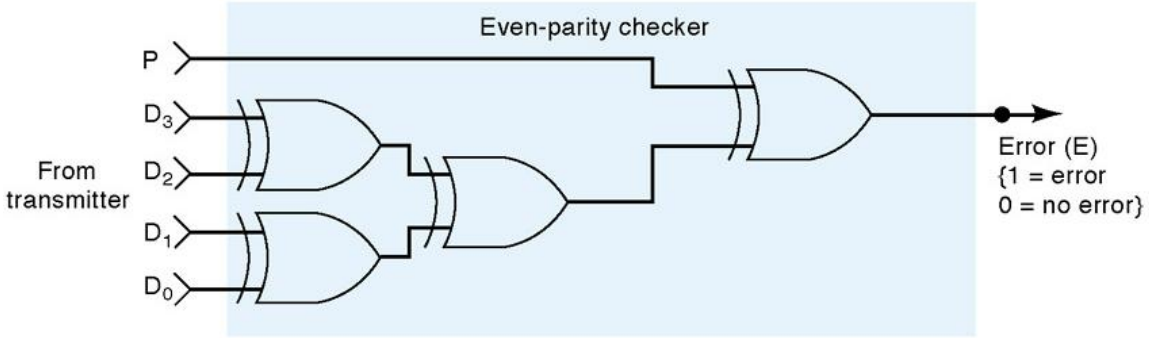
Parity Generator and Checker

Similarly for 4 bits:



(a)

Even Parity:



(b)

Enable/Disable Circuits

- A circuit is enabled when it allows the passage of an input signal to the output.
- A circuit is disabled when it prevents the passage of an input signal to the output.
- Situations requiring enable/disable circuits occur frequently in digital circuit design.

FIGURE 4-26 Four basic gates can either enable or disable the passage of an input signal, *A*, under control of the logic level at control input *B*.