

# DIGITAL SYSTEMS AND MICROPROCESSORS (ELE2002M)

LECTURE 3 - GATE LEVEL MINIMIZATION (CONT...)

Instructor:

Assoc. Prof. Dr. Edmond Nurellari

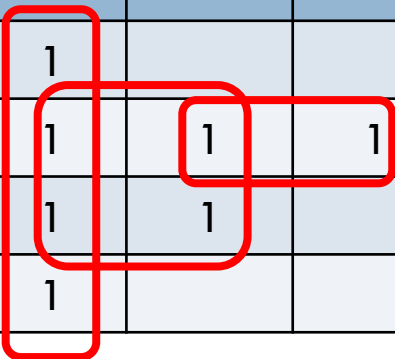
| CD\AB | 00 | 01 | 11 | 10 |
|-------|----|----|----|----|
| 00    | 1  | 1  | 1  | 1  |
| 01    |    |    |    |    |
| 11    |    |    |    |    |
| 10    | 1  | 1  | 1  | 1  |

$$F=D'$$

Implement the following function in both NAND and NOR form  
[Hint: for NOR form implement the  $F'$ ]

$$F(A,B,C,D) = (4,5,6,7,9,13,15)$$

| CD\AB | 00 | 01 | 11 | 10 |
|-------|----|----|----|----|
| 00    |    | 1  |    |    |
| 01    |    | 1  | 1  | 1  |
| 11    |    | 1  | 1  |    |
| 10    |    | 1  |    |    |



The Karnaugh map shows the function F(A,B,C,D) with minterms 4, 5, 6, 7, 9, 13, and 15. The map is a 4x4 grid with columns labeled 00, 01, 11, 10 and rows labeled 00, 01, 11, 10. Red lines group the 1s into three prime implicants: a vertical group of four 1s in the 01 column (minterms 4, 5, 6, 7), a horizontal group of three 1s in the 01 row (minterms 5, 6, 7), and a horizontal group of two 1s in the 11 row (minterms 6, 7).



Implement the following function in both NAND and NOR form  
[Hint: for NOR form implement the  $F'$ ]

$$F(A,B,C,D) = (4,5,6,7,9,13,15)$$

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

# Other Two-Level Implementations

---

- Wired AND (Also called AND-OR-Invert )
- Wired OR (Also called OR-AND-Invert)

# AND-OR-Invert Implementation

---

$$F = (AB + CD + E)'$$

# OR-AND-Invert Implementation

---

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



# Homework Exercise

Implement  $F' = x'y + xy' + z$  using

1. AND-NOR and NAND-AND (using AND-OR-Invert)
2. OR-NAND and NOR-OR (using OR-AND-Invert)

# Exclusive-OR Function

- The exclusive-OR (XOR) is represented by the symbol  $\oplus$
- The exclusive-OR of 2 variables (x,y) is equal to 1 only if  $x = 1$  OR  $y=1$  (but not both)
- Similarly XNOR of 2 variables (x,y) is equal to 1 only if  $x = 0$  OR  $y = 0$  (but not both)

$$a \text{ xor } 0 = a$$

$$a \text{ xor } 1 = a'$$

$$a \text{ xor } a = 0$$

$$a \text{ xor } a' = 1$$

$$a \text{ xor } b' = a' \text{ xor } b = (a \text{ xor } b)'$$

$$a \text{ xor } b = b \text{ xor } a$$

$$(a \text{ xor } b) \text{ xor } c = a \text{ xor } (b \text{ xor } c) = a \text{ xor } b \text{ xor } c$$

# Odd function

- For a 2 variable odd function only 1 variable must be equal to 1
- A multiple-variable exclusive-OR function is defined as an **odd function**
- Generally, if the minterms of a function consist of only odd number of 1s it is an odd function
- An n-variable exclusive-OR function is an odd function defined as the logical sum of  $2^{n-1}$  minterms whose binary numerical values have an odd number of 1's

# Parity Generation and Checking



Reading Assignment

# Combinational Logic

- Combinational logic (CL) consists of input variables, logic gates and output variables.
- CL reacts to the values of signals at their inputs and produce a value of output signal.
- For  $n$ -input variables, there are  $2^n$  possible binary input combinations.

# Analysis Procedure



Reading Section 4.3

# Design Procedure



Reading Assignment Section 4.4

# Binary Adder

- The half adder
  - ▣ The half adder is a circuit for adding two single bit numbers
  - ▣ Develop a truth table and Boolean expressions for the half adder

| A | B | S | C |
|---|---|---|---|
| 0 | 0 |   |   |
| 0 | 1 |   |   |
| 1 | 0 |   |   |
| 1 | 1 |   |   |

S and C are the Sum and Carry



# Half adder

- The sum is XOR operation and the carry an AND:

| A | B | S | C |
|---|---|---|---|
| 0 | 0 |   |   |
| 0 | 1 |   |   |
| 1 | 0 |   |   |
| 1 | 1 |   |   |

