Logic Gates (2)

Lecture 09

COE211-Digital Logic Design

الفصل الدراسي الأول 1442-Fall2020

جامعة طيبة فرع ينبع - كلية علوم وهندسة الحاسبات – شطر الطالبات

د. فاطمة الحربي

Midterm

- **The midterm is scheduled to be on Wednesday 21/10/2020 from 9:30AM to 10:30AM**
- The schedule can be changed in two cases:
  - A student has more than 2 midterms on the same day
  - A student has 2 exams at the same time

Universal Gates: NAND and NOR

- AND/OR/NOT gates are sufficient for building any Boolean functions
- However, other gates are also used because:
  - (i) usefulness
  - (ii) economical on transistors
  - (iii) self-sufficient

NAND/NOR: economical, self-sufficient

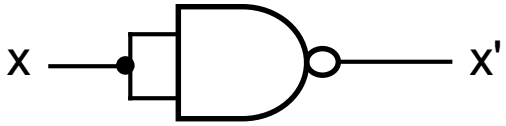
XOR: useful (e.g. parity bit generation)

NAND Gate

- NAND gate is **self-sufficient** (**you can build any logic circuit with it**)
- Can be used to implement AND/OR/NOT

Implementations using NAND Gate (1)

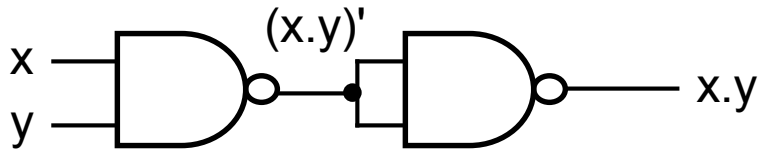
- Implementing an inverter using NAND gate:



(x.x)' = x' (T1: idempotency)

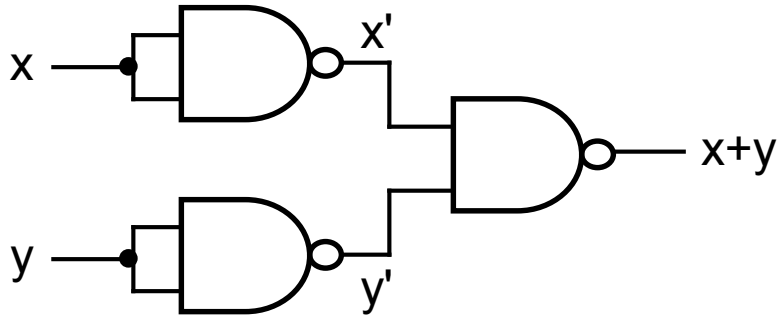
Implementations using NAND Gate (2)

- Implementing AND using NAND gates:



((xy)'(xy)')' = ((xy)')' idempotency
= (xy) involution

- Implementing OR using NAND gates:



((xx)'(yy)')' = (x'y')' idempotency
= x''+y'' DeMorgan
= x+y involution

Implementations using NAND Gate (3)

- Possible to implement any Boolean expression using NAND gates
- *Procedure:*
  1. Obtain sum-of-products Boolean expression:

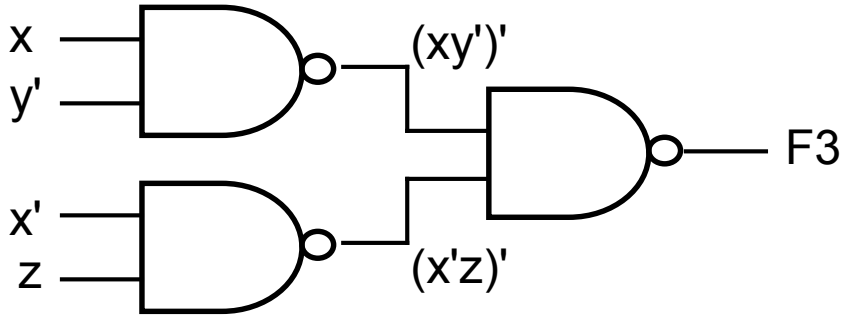
     (e.g.,) F3 = xy'+x’z
  2. Use DeMorgan theorem to obtain expression using 2-level NAND gates:

     (e.g.,) F3 = xy'+x'z

     = (xy'+x'z)' ' involution

     = ((xy')' . (x'z)')' DeMorgan

Implementations using NAND Gate (4)



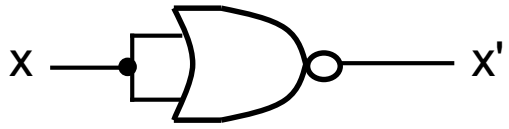
F3 = ((xy')'.(x'z)') ' = xy' + x'z

NOR Gate

- NOR gate is also self-sufficient
- Can be used to implement AND/OR/NOT

Implementations using NOR Gate (1)

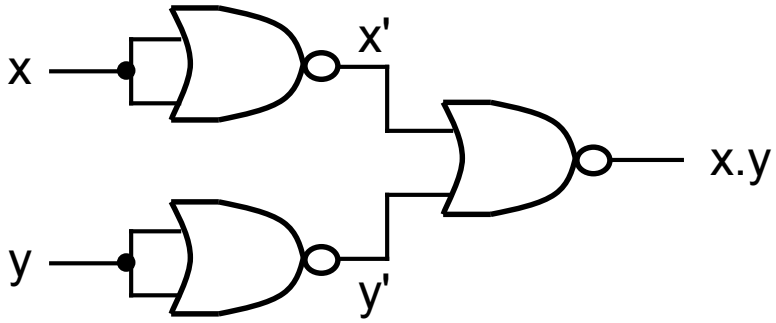
- Implementing an inverter using NOR gate:



$(x+x)' = x'$ (T1: idempotency)

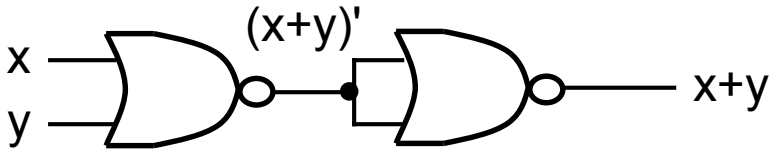
Implementations using NOR Gate (2)

- Implementing AND using NOR gates:



((x+x)'+(y+y)')'=(x'+y')' idempotency
= x''.y'' DeMorgan
= x.y involution

- Implementing OR using NOR gates:



((x+y)'+(x+y)')' = ((x+y)')' idempotency
= (x+y) involution

Implementations using NOR Gate (3)

- Possible to implement boolean expression using NOR gates
- *Procedure:*
  1. Obtain product-of-sums Boolean expression:

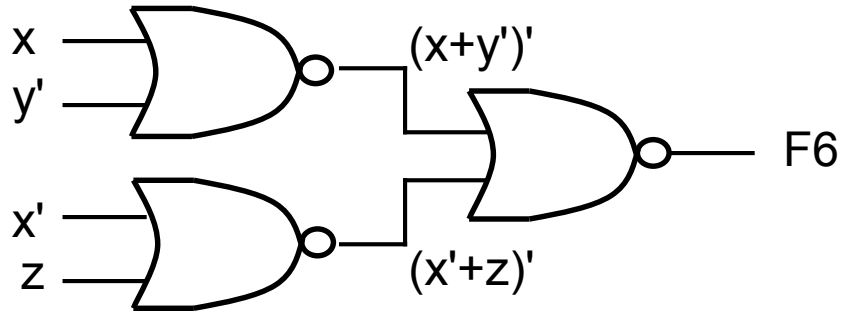
     (e.g.,) F6 = (x+y').(x’+z)
  2. Use DeMorgan theorem to obtain expression using 2-level NOR gates:

(e.g.,) F6 = (x+y').(x'+z)

= ((x+y').(x'+z))' ' involution

= ((x+y')'+(x'+z)')' DeMorgan

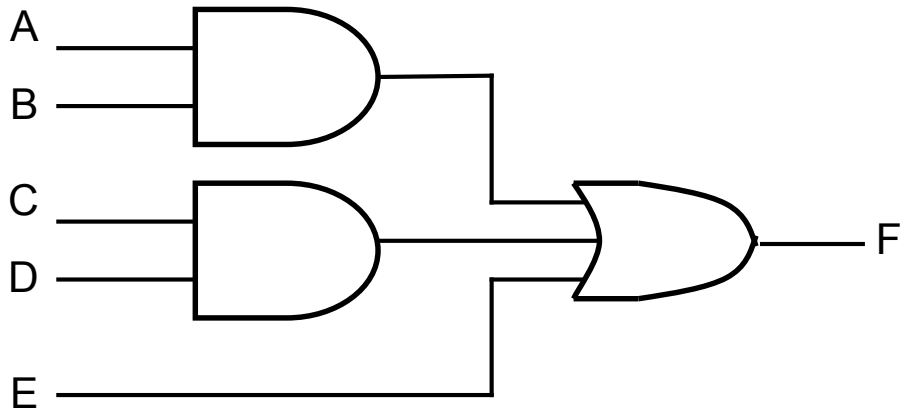
Implementations using NOR Gate (4)



F6 = ((x+y')'+(x'+z)')' = (x+y').(x'+z)

Implementation of SOP Expressions (1)

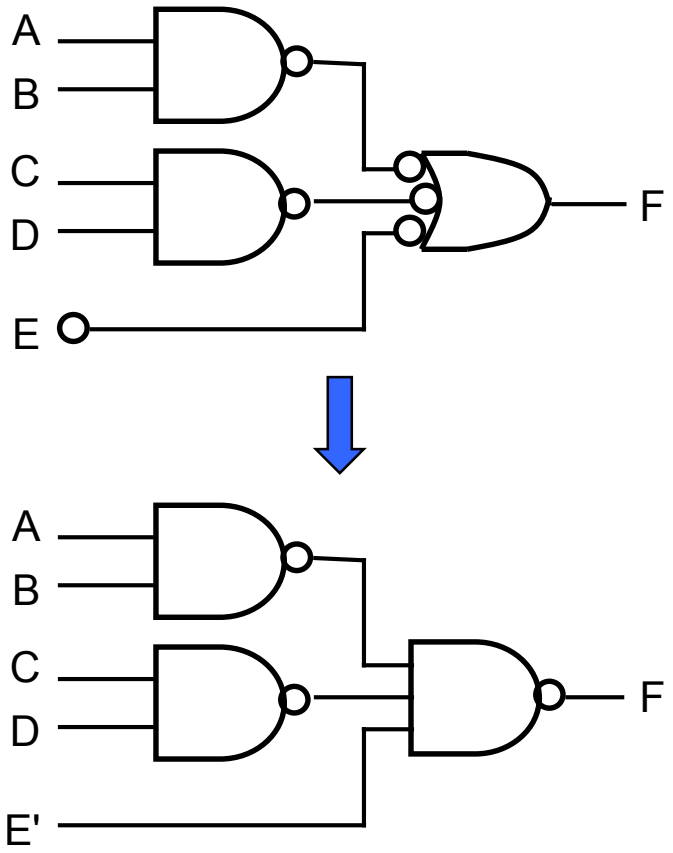
- Sum-of-Products (SOP) expressions can be implemented using:
  - 2-level AND-OR logic circuits
  - 2-level NAND logic circuits
- AND-OR logic circuit



F = AB + CD + E

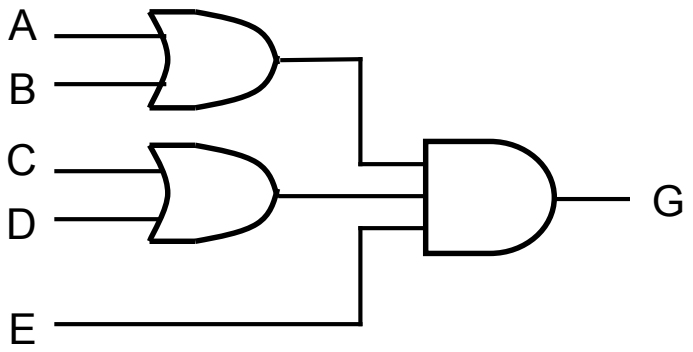
Implementation of SOP Expressions (2)

- NAND-NAND circuit (by circuit transformation):
  a) add double bubbles
  b) Change OR-with-inverted-inputs to NAND & bubbles at inputs to their complements



Implementation of POS Expressions (1)

- Product-of-Sums expressions can be implemented using:
  - 2-level OR-AND logic circuits
  - 2-level NOR logic circuits
- OR-AND logic circuit



G = (A+B).(C+D).E

Implementation of POS Expressions (2)

- NOR-NOR circuit (by circuit transformation):
  a) add double bubbles
  b) changed AND-with-inverted-inputs to NOR & bubbles at inputs to their complements