Spring 2015 Week 2 Module 9

Digital Circuits and Systems

K-Map Minimization

*Shankar Balachandran**
Associate Professor, CSE Department
Indian Institute of Technology Madras

*Currently a Visiting Professor at IIT Bombay

K-Map Based Minimization

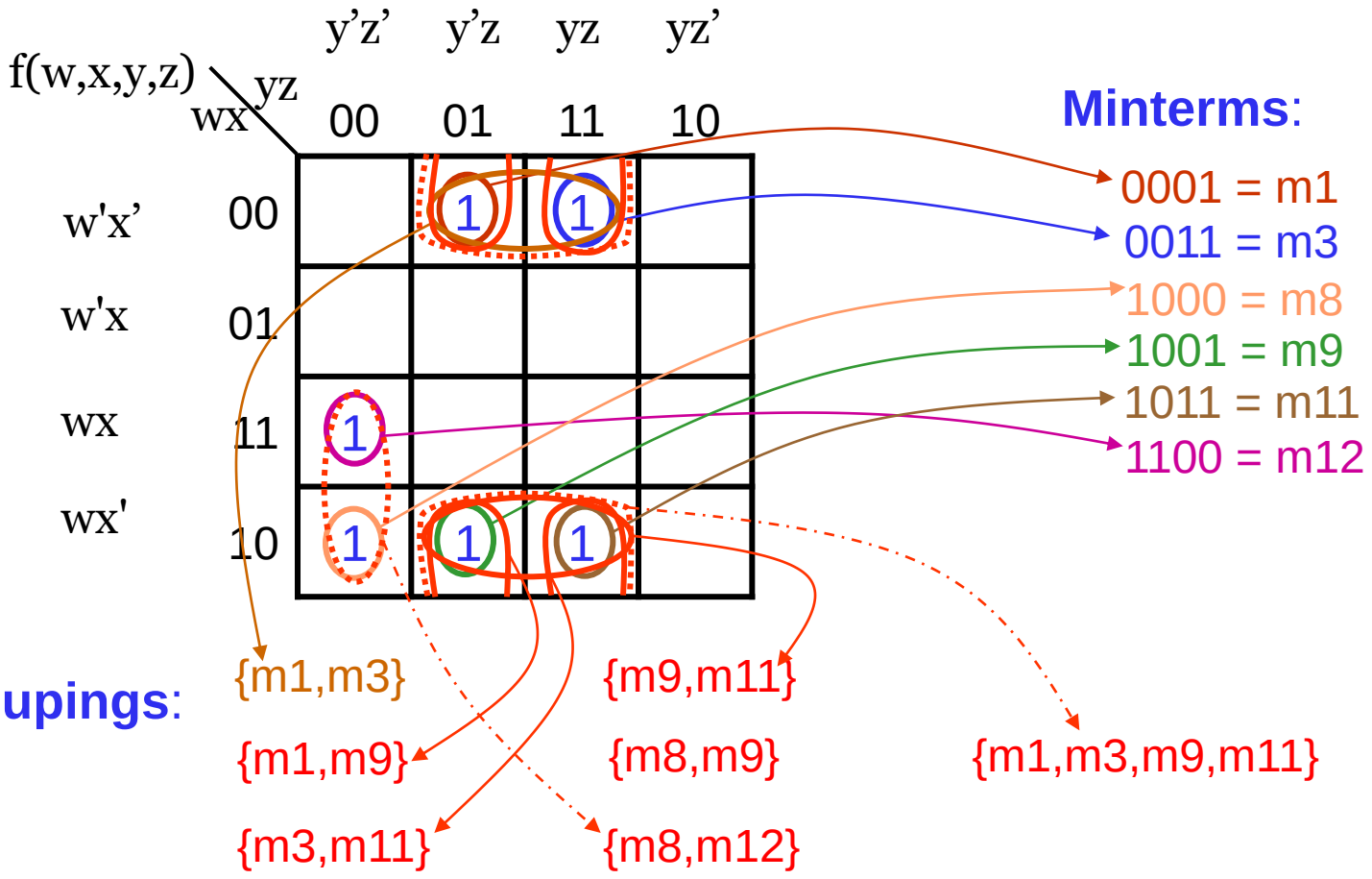
- The purpose of using K-maps is to aid in the simplification of logic expressions.
- Use of 1's in K-maps gives minterm SOP form directly. Use of 0's in K-maps gives maxterm POS form directly.
- Minterms (maxterms) that can be combined differ in *exactly* one bit position.
- Minimization using K-maps is done by grouping "*adjacent*" minterms (maxterms).

- **Conditions for grouping:**
  - Minterm (maxterm) groups are restricted to have size that is a power of 2, e.g., 1, 2, 4, 8, …
  - All K-map cells in a group must have a 1 as their K-map entry for grouping minterms (or 0 as their K-map entry for maxterm groups).
  - All minterms (maxterms) must be adjacent; they must *differ* in *exactly* $\boldsymbol{i}$ *bits* where, $\mathbf{2^i}$ is the size of the group.

- To minimize a function, find the set of groups which include all the required minterms; the function is the sum of these groups (or if maxterms are used then the function is the product of the groups).

- The larger the groups, the better. *Why?*

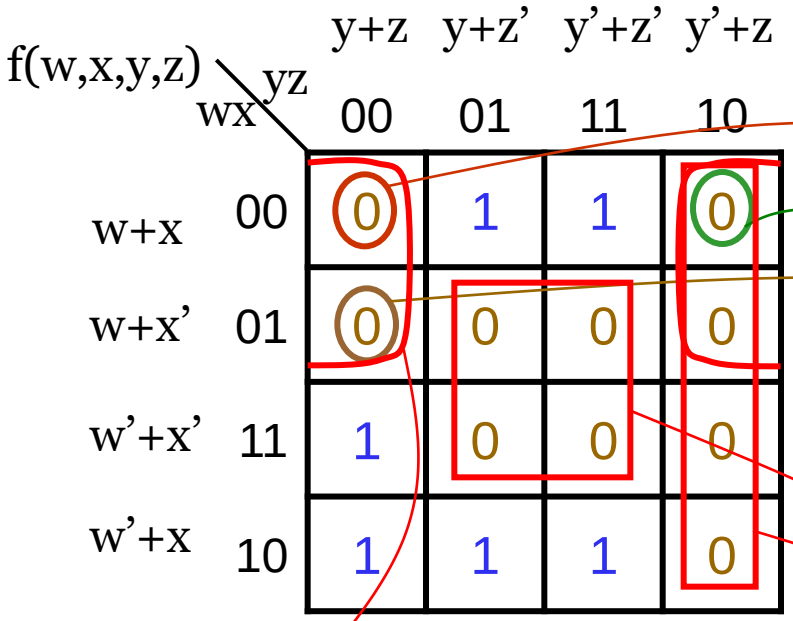
Example: Grouping Minterms



Simplified Expression: $f(w,x,y,z) = \{m1,m3,m9,m11\} + \{m8,m12\}$

$$= \bar{x}z + w\bar{y}\bar{z}$$

Example: Grouping Maxterms



Maxterms:

0000 = M0
0010 = M2
0100 = M4
0101 = M5
0110 = M6
0111 = M7
1010 = M10
1101 = M13
1110 = M14
1111 = M15

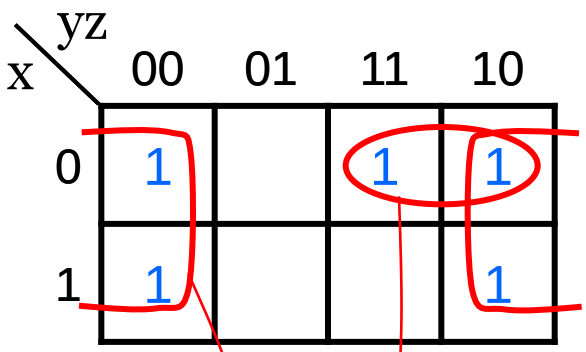
Groupings (note: all groupings are not listed):

{M0,M2,M4,M6}, {M4,M5,M6,M7}, {M5,M7,M13,M15},
{M2,M6,M10,M14}, {M6,M7,M14,M15}

Simplified Expression: $f(w,x,y,z) = \{M0,M2,M4,M6\}\{M5,M7,M13,M15\}\{M2,M6,M10,M14\}$

$$= (w+z)(\overline{x}+\overline{z})(\overline{y}+z)$$

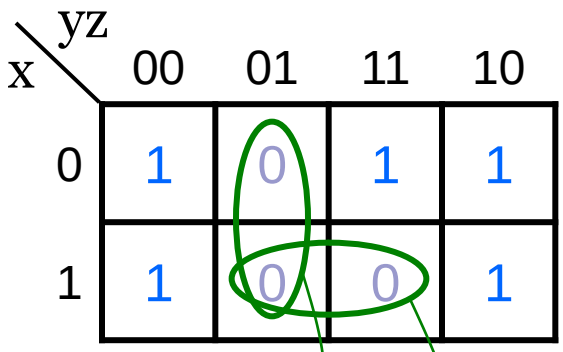
Logic Minimization: Example 1



Sum of Products:

$$F(x,y,z) = \bar{z} + \bar{x}y$$

INV = 2
AND2 = 1
OR2 = 1



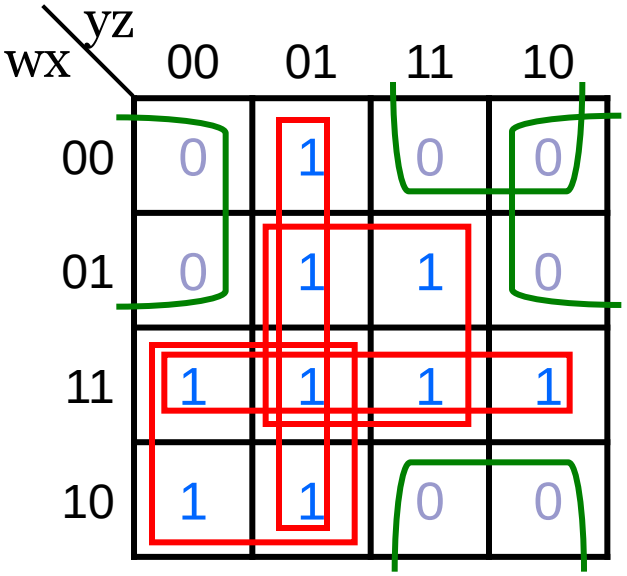
Product of Sums:

$$F(x,y,z) = (y+\bar{z})(\bar{x}+\bar{z})$$

INV = 2
AND2 = 1
OR2 = 2

⇨ **For example 1 SOP is SMALLER**

Logic Minimization: Example 2



SOP:

$$F(w,x,y,z)=wx+w\bar{y}+\bar{y}z+xz$$

INV = 1
AND2 = 4
OR2 = 3

POS:

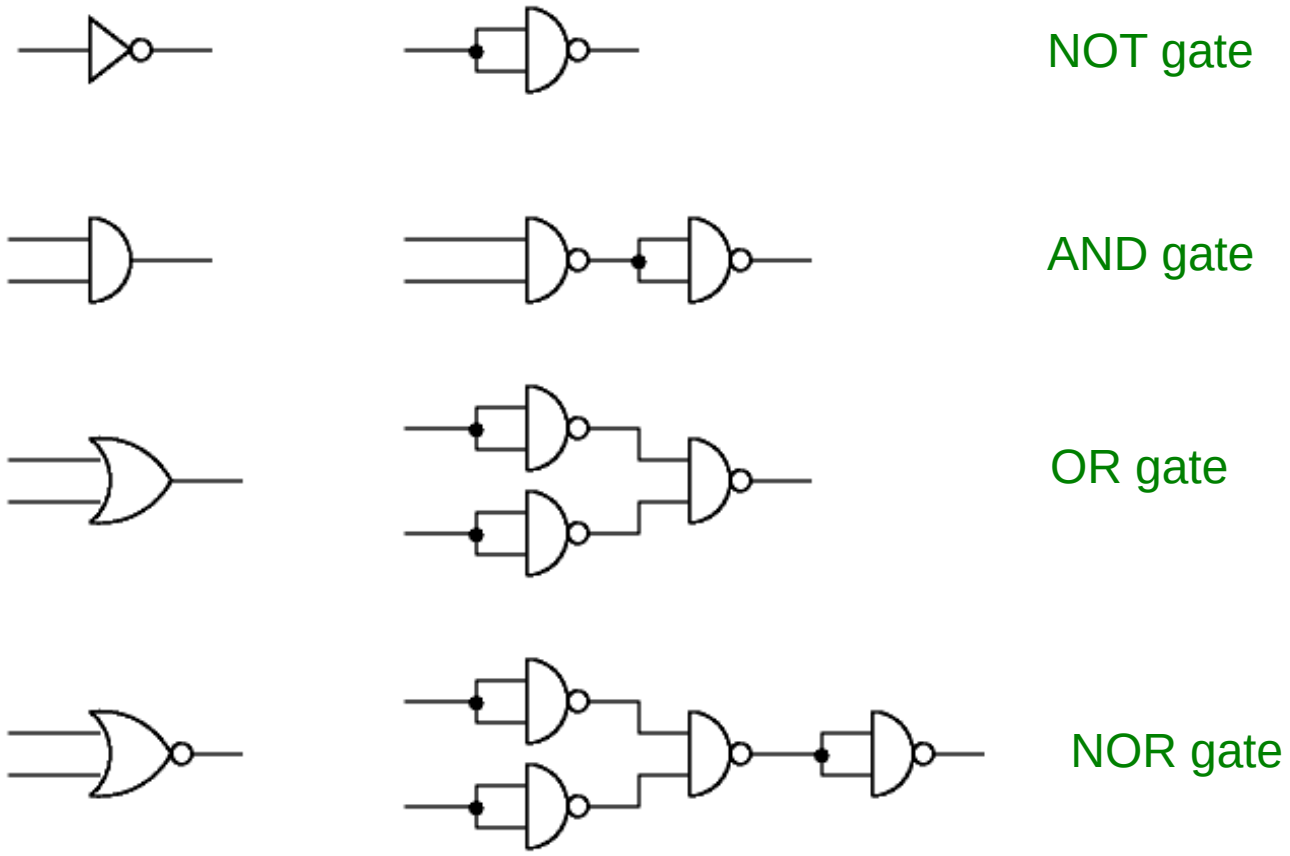
$$F(w,x,y,z)=(w+z)(x+\bar{y})$$

INV = 1
AND2 = 1
OR2 = 2

⇨ **For example 2 POS is SMALLER**

Gate Equivalents

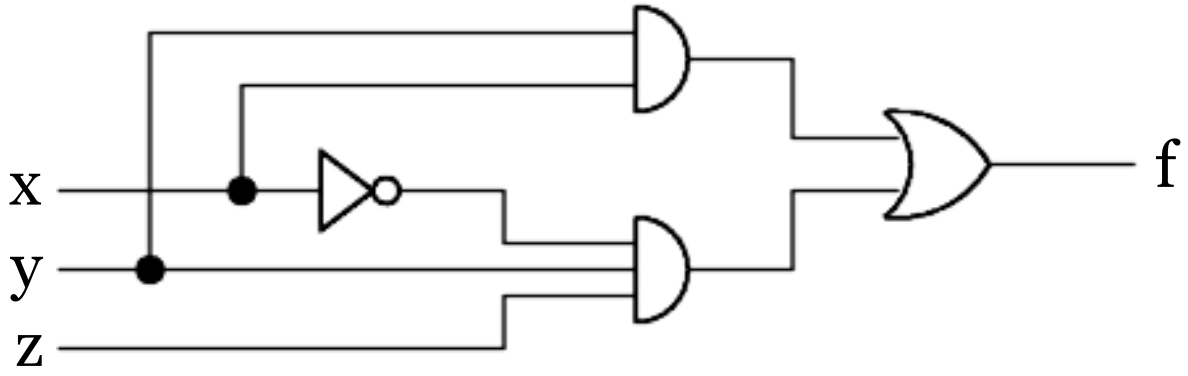
- Gate equivalents using 2-input NAND gates



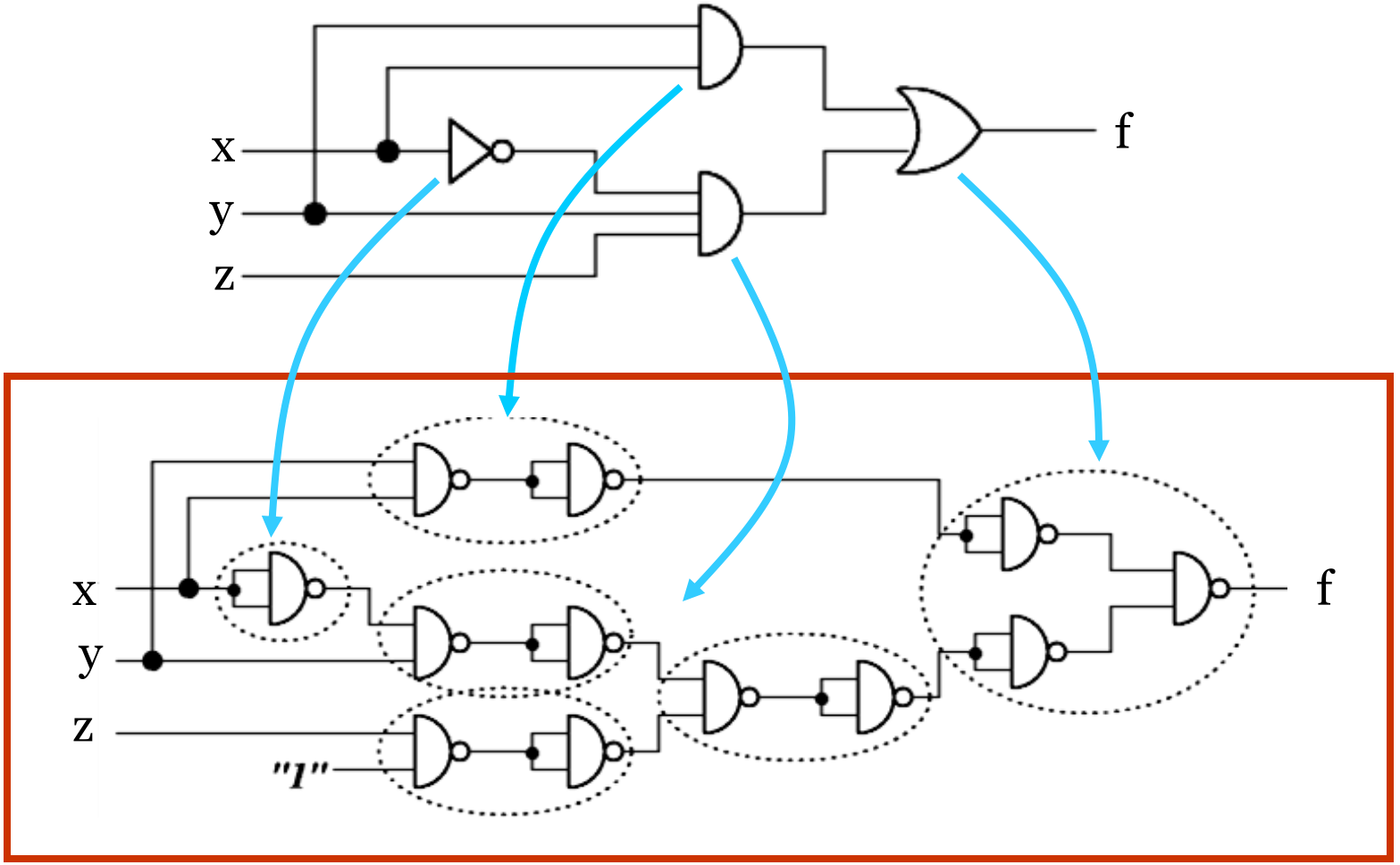
Example:

Implement $(f(x,y,z) = xy + \overline{x}yz)$ using only 2-input NAND gates.

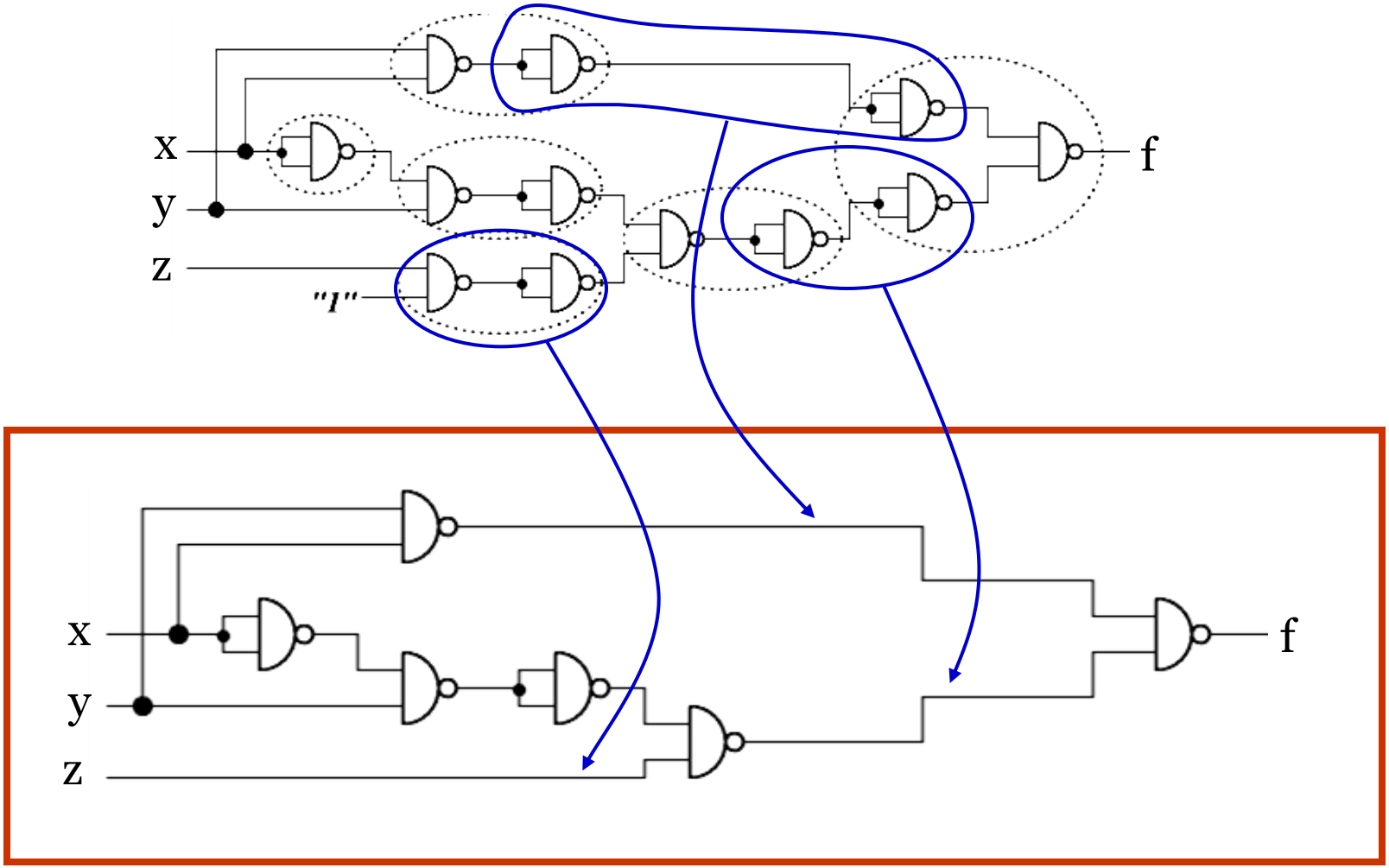
1. Create an AND/OR circuit



2. Substitute all logic gates by equivalent 2-input NAND gates



3. Remove redundant NAND gates



End of Week 2: Module 9

Thank You