JOINT INSTITUTE
交大密西根学院

ECE2700J SU23 RC3

Logic Optimization, K-map

Wenyue Li
6/3/2024

Logic Optimization

Simplification and Optimization

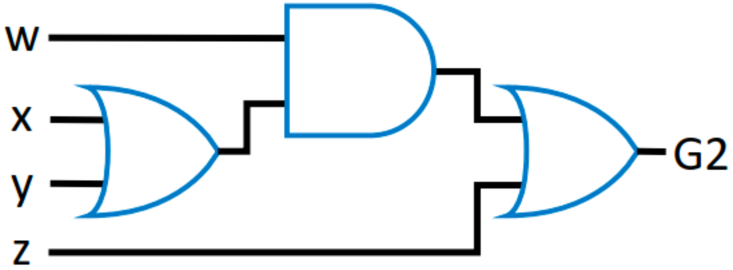
Intention : design a **better** circuit

– Delay – the time from input change to correct stable output response

- Every gate has delay of “1 gate-delay”, the gate delay of a circuit is judged by its critical path

– Size – the number of transistors

- Every gate input requires 2 transistors



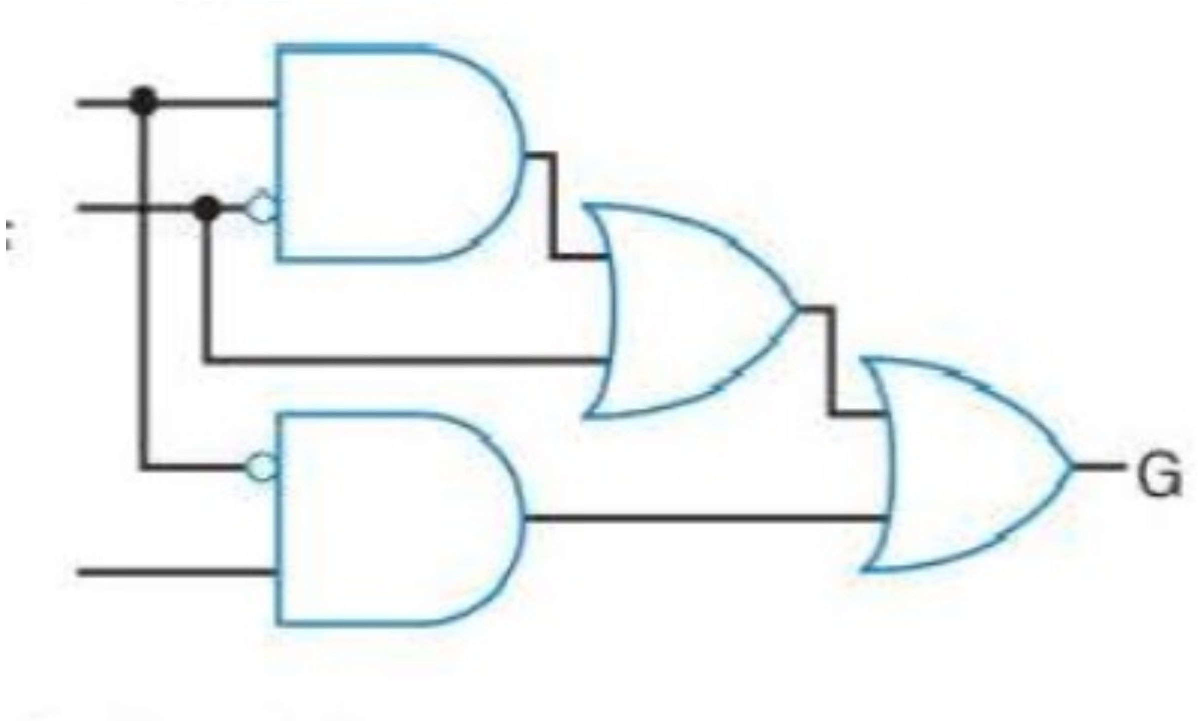
- Ignore inverters for simplicity

– Sacrifices delay for smaller size

Logic Optimization

Simplification and Optimization

Exercise: find the sizes and Delay of the following circuit



Logic Optimization

Simplification and Optimization

Exercise: Optimize the circuit with the equation shown below. Write its original size and delay and the optimized sizes and delay.

$$F = xyz + xyz' + x'y'z' + x'y'z$$

K-map

- A graphical technique used to simplify a logic equation

- A way to show the relationship between the logic inputs and corresponding output
  – Like truth table

- Much cleaner and more procedural than algebraic simplification by theorems of Boolean algebra.

- Theoretically, it can be used for any number of input variables,

– BUT is only practical for less than six, we will limit our discussion to logic equations with five or less variables

K-map

How to draw K-map

- K-map can be filled up directly from a truth table

– Each minterm corresponds to a cell in the K-map

- K-map cells are labeled so that both horizontal and vertical movement differ only in one variable

F X\Y	Y'	Y
X'	m0	m1
X	m2	m3

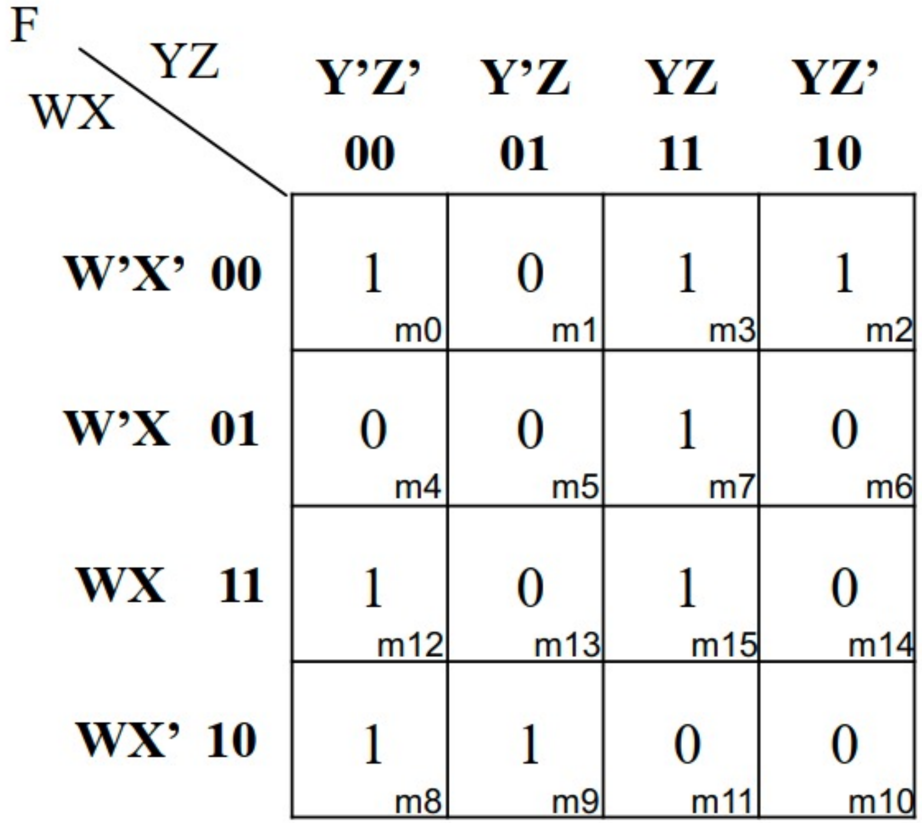
F X\YZ	Y'Z'	Y'Z	YZ	YZ'
X'	m0	m1	m3	m2
X	m4	m5	m7	m6

F WX\YZ	Y'Z'	Y'Z	YZ	YZ'
W'X'	m0	m1	m3	m2
W'X	m4	m5	m7	m6
WX	m12	m13	m15	m14
WX'	m8	m9	m11	m10

K-map

Basic knowledge of K-map

Label the Rows and Columns by 0 and 1



F WX \ YZ	Y’Z’ 00	Y’Z 01	YZ 11	YZ’ 10
W’X’ 00	1 m0	0 m1	1 m3	1 m2
W’X 01	0 m4	0 m5	1 m7	0 m6
WX 11	1 m12	0 m13	1 m15	0 m14
WX’ 10	1 m8	1 m9	0 m11	0 m10

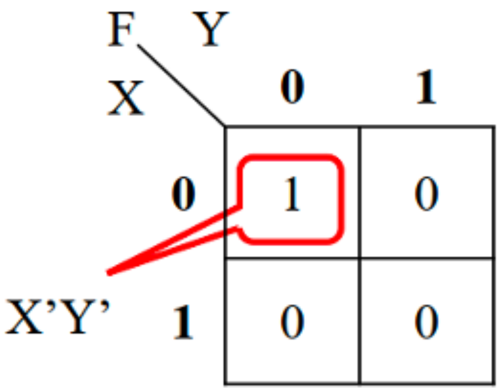
We can write a equation from K-map similar to truth table.

F=

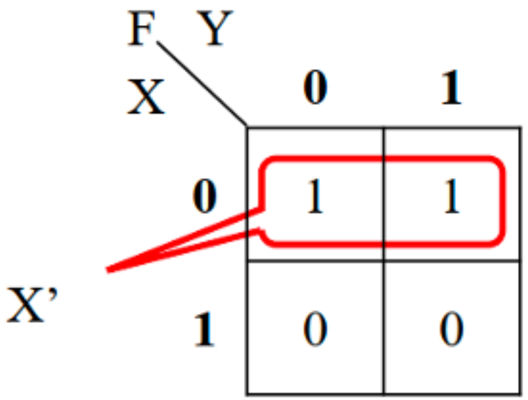
K-map

Grouping and Cancelling

Why we need to group?



	0	1
0	1	0
1	0	0



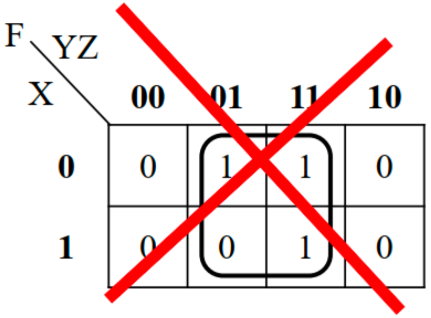
	0	1
0	1	1
1	0	0

K-map

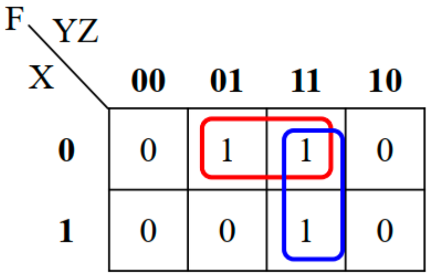
Grouping and Cancelling

Grouping rules

- No zeros in the group

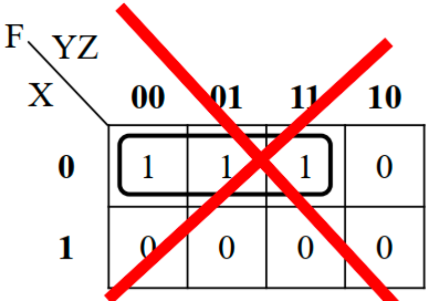


F X \ YZ	00	01	11	10
0	0	1	1	0
1	0	0	1	0

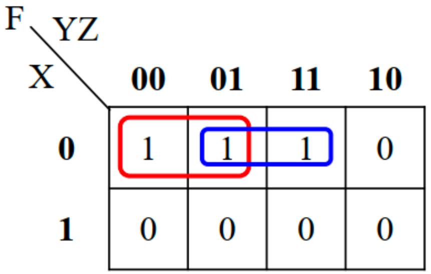


F X \ YZ	00	01	11	10
0	0	1	1	0
1	0	0	1	0

- The number of 1's in a group should be 2^N, N = 0, 1, 2, …

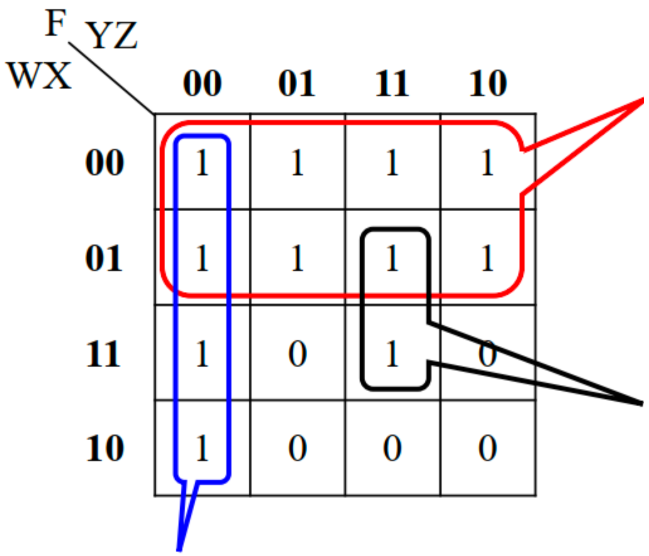


F X \ YZ	00	01	11	10
0	1	1	1	0
1	0	0	0	0



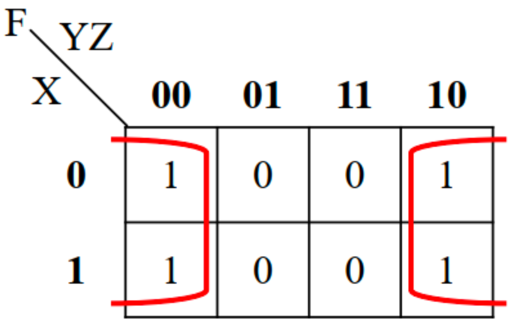
F X \ YZ	00	01	11	10
0	1	1	1	0
1	0	0	0	0

- Group as many adjacent 1's as possible



F WX \ YZ	00	01	11	10
00	1	1	1	1
01	1	1	1	1
11	1	0	1	0
10	1	0	0	0

- Edges wrap around

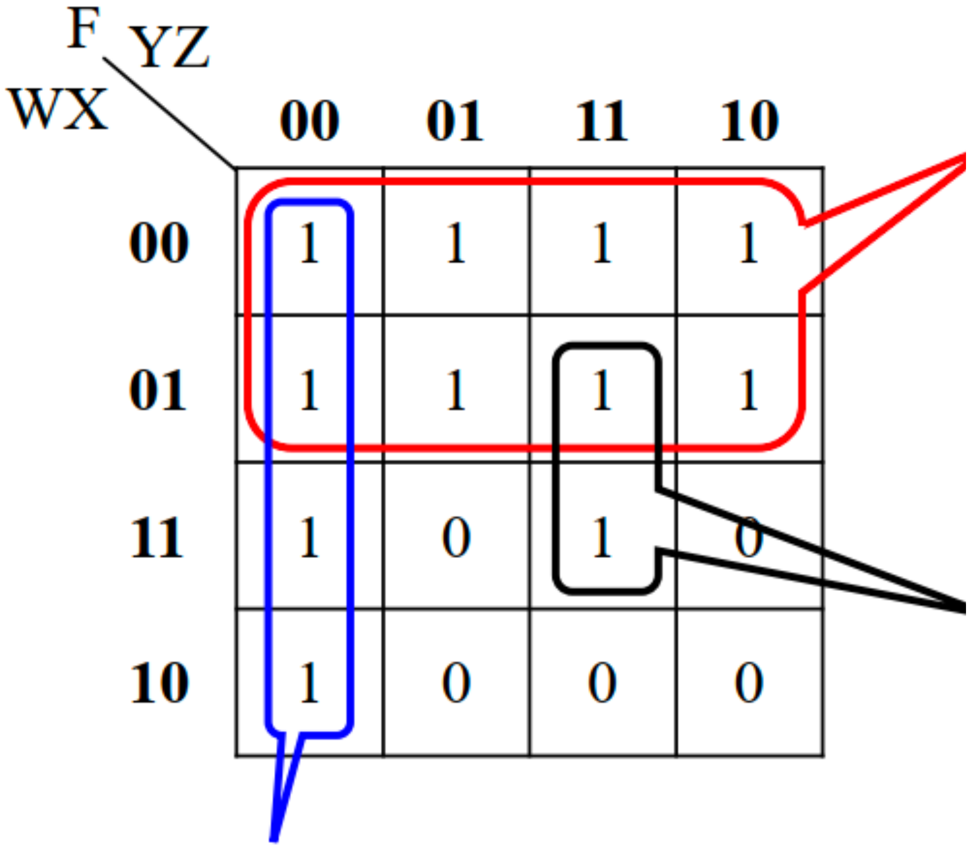


F X \ YZ	00	01	11	10
0	1	0	0	1
1	1	0	0	1

K-map

Grouping and Cancelling

How to quickly use group to simplify a circuit?

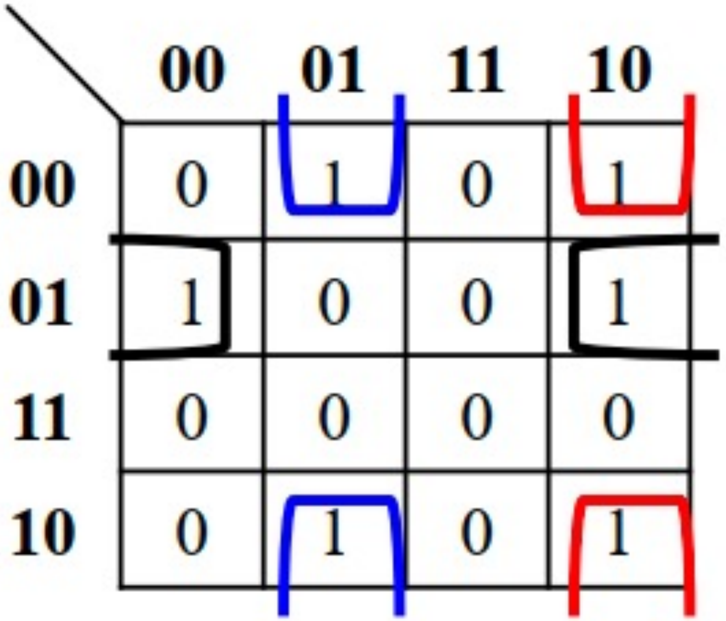


F YZ WX	00	01	11	10
00	1	1	1	1
01	1	1	1	1
11	1	0	1	0
10	1	0	0	0

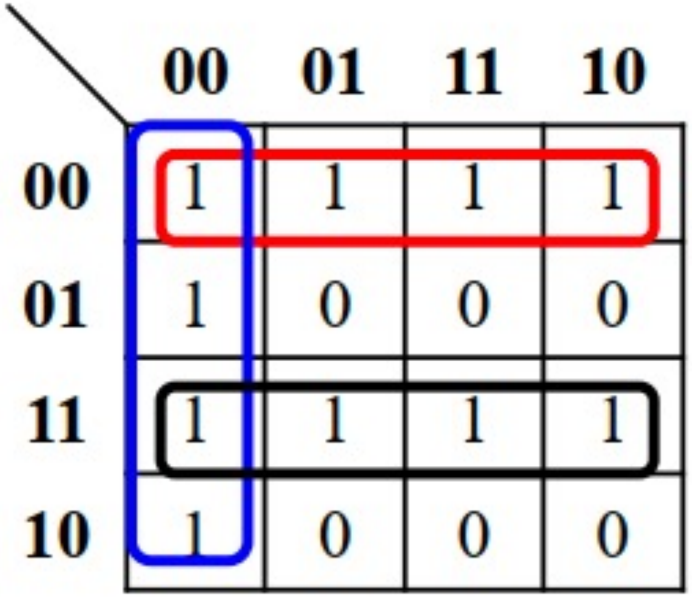
K-map

Grouping and Cancelling

Exercise: Quickly write the equation of the circled groups.



	00	01	11	10
00	0	1	0	1
01	1	0	0	1
11	0	0	0	0
10	0	1	0	1



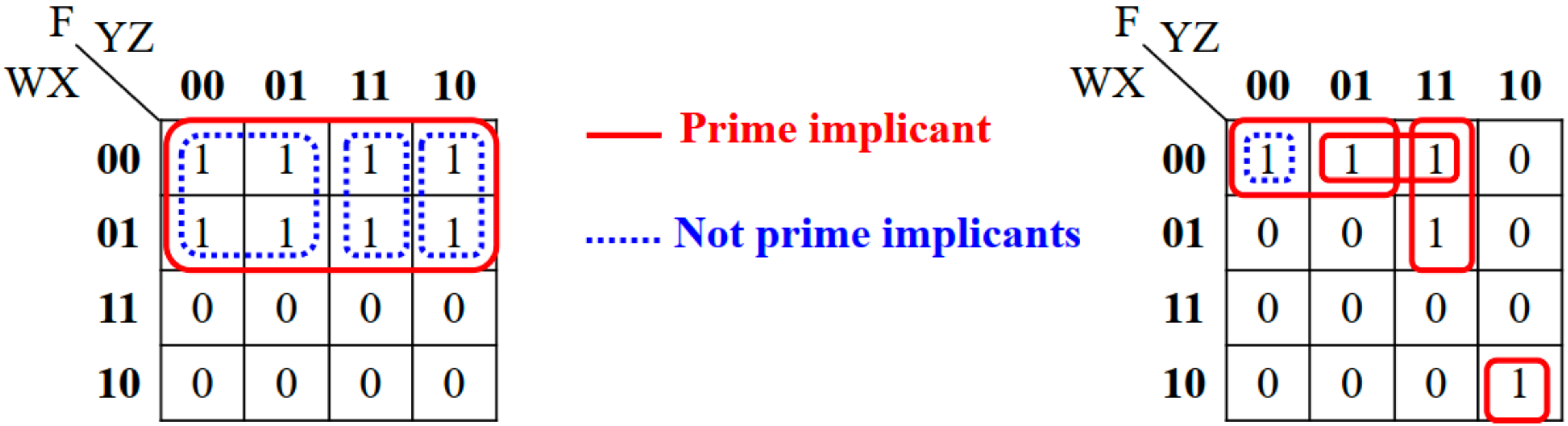
	00	01	11	10
00	1	1	1	1
01	1	0	0	0
11	1	1	1	1
10	1	0	0	0

K-map

Prime Implicants

- Implicant: is a product term

- A prime implicant (PI) is a group that cannot be entirely contained by another implicant



F WX \ YZ	00	01	11	10
00	1	1	1	1
01	1	1	1	1
11	0	0	0	0
10	0	0	0	0

— Prime implicant

······· Not prime implicants

F WX \ YZ	00	01	11	10
00	1	1	1	0
01	0	0	1	0
11	0	0	0	0
10	0	0	0	1

K-map

Prime Implicant

Exercise: Find all PIs in the following K-map

F WX \ YZ	00	01	11	10
00	1	0	1	1
01	0	1	1	0
11	0	1	1	0
10	1	1	1	1

K-map

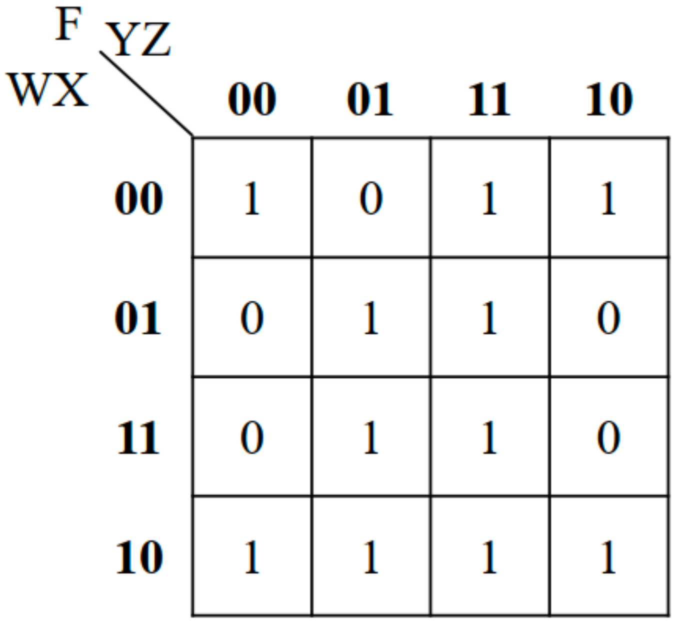
Essential Prime Implicant

- A prime implicant (PI) is essential if a a cell is covered ONLY by that PI

- The essential PIs can be found by

– looking at each cell marked as 1 and not covered by any other essential PI

– and checking the number of PIs that cover it

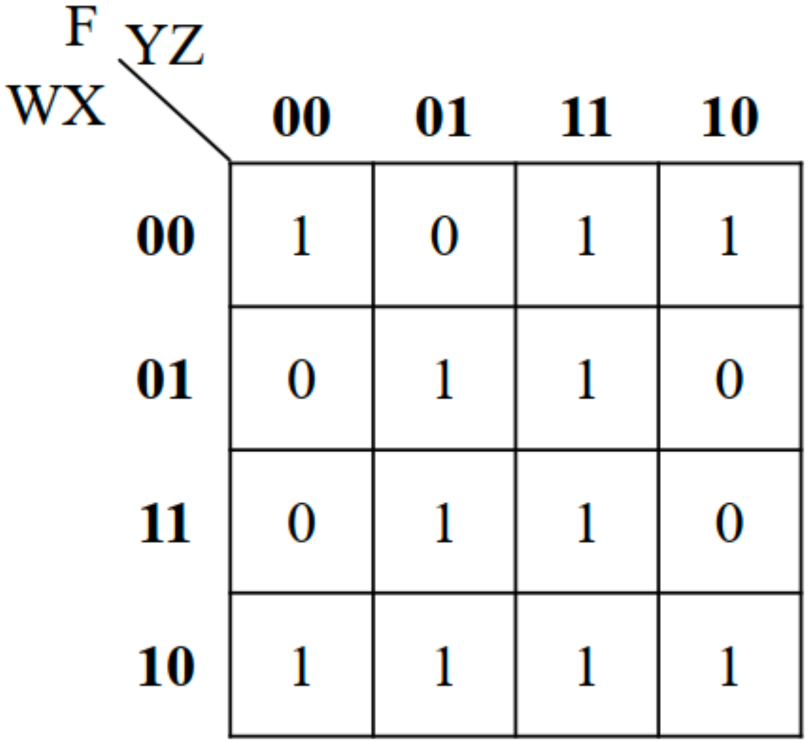


F YZ WX	00	01	11	10
00	1	0	1	1
01	0	1	1	0
11	0	1	1	0
10	1	1	1	1

K-map

Essential Prime Implicant

- Essential PIs **have to be used in the simplified equation**
- Cells not covered by essential PIs can be represented by any PIs covering them

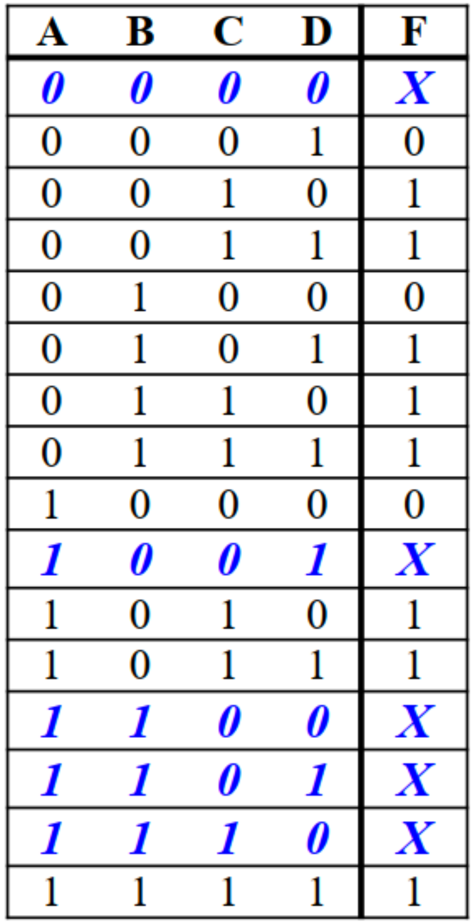


F WX \ YZ	00	01	11	10
00	1	0	1	1
01	0	1	1	0
11	0	1	1	0
10	1	1	1	1

K-map

Don’t care conditions in K-map

- The possible input combinations might not be all valid or not for consideration for a device
  - Hence we don’t care what the corresponding outputs are under those conditions
  - Called **don’t care** conditions
  - Mark the corresponding outputs by **X**



A	B	C	D	F
0	*0*	*0*	*0*	*X*
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	*0*	*0*	*1*	*X*
1	0	1	0	1
1	0	1	1	1
1	*1*	*0*	*0*	*X*
1	*1*	*0*	*1*	*X*
1	*1*	*1*	*0*	*X*
1	1	1	1	1