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

Boolean Algebra - Basic Rules

1. $A + 0 = A$
2. $A + 1 = 1$
3. $A \bullet 0 = 0$
4. $A \bullet 1 = A$
5. $A + A = A$
6. $A + \overline{A} = 1$
7. $A \bullet A = A$
8. $A \bullet \overline{A} = 0$
9. $\overline{\overline{A}} = A$
10. $A + AB = A$
11. $A + \overline{A}B = A + B$
12. $(A + B)(A + C) = A + BC$

Simplification from looping:

Pair: Looping a pair of adjacent 1's eliminate the variable that appears in complemented and uncomplemented form.

Simplification from looping:

Pair: Looping a pair of adjacent 1's eliminate the variable that appears in complemented and uncomplemented form.

Quad: Looping 4 adjacent 1's eliminate the two variables that appears in complemented and uncomplemented form.

Simplification from looping:

Pair: Looping a pair of adjacent 1's eliminate the variable that appears in complemented and uncomplemented form.

Quad: Looping 4 adjacent 1's eliminate the two variables that appears in complemented and uncomplemented form.

Octet: Looping 8 adjacent 1's eliminate the three variables that appears in complemented and uncomplemented form.

Complete K-Map simplification process

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) even if they have already been looped.
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to include 1s not already looped.
7. Form the OR sum of terms generated by each loop.

Example (a) K-Map simplification

Simplify the following Boolean expression:

$\bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}C\bar{D} + ABCD + \bar{A}BCD + A\bar{B}CD +$

$ABCD$

Example (a) K-Map simplification

Simplify the following Boolean expression:

$\overline{A}\,\overline{B}\,\overline{C}D + \overline{A}\,\overline{B}C\overline{D} + ABCD + \overline{A}BCD + A\overline{B}CD +$

$ABCD$

Sum of Product (SOP) expression

Example (a) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}C\bar{D} + ABCD + \bar{A}BCD + A\bar{B}CD + ABCD$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$				
$\bar{A}B$				
AB				
$A\bar{B}$				

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (a) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}\bar{C}D + \bar{A}BC\bar{D} + ABCD + \bar{A}BCD + A\bar{B}CD + ABCD$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$				1
$\bar{A}B$		1	1	
AB		1	1	
$A\bar{B}$			1	

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (a) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}\bar{C}D + \bar{A}BC\bar{D} + ABCD + ABCD + A\bar{B}CD +$$

$$ABCD$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\,\bar{B}$				1
$\bar{A}B$		1	1	
AB		1	1	
$A\bar{B}$			1	

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (a) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}\bar{C}D + \bar{A}BC\bar{D} + ABCD + ABCD + A\bar{B}CD + ABCD$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\,\bar{B}$				1
$\bar{A}B$		1	1	
AB		1	1	
$A\bar{B}$			1	

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (a) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}C\bar{D} + \bar{A}B\bar{C}D + \bar{A}BCD + AB\bar{C}D + A\bar{B}CD + ABCD$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$				1
$\bar{A}B$		1	1	
AB		1	1	
$A\bar{B}$			1	

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped. (none here)*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (a) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}\bar{C}D + \bar{A}BC\bar{D} + ABCD + ABCD + A\bar{B}CD + ABCD$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$				1
$\bar{A}B$		1	1	
AB		1	1	
$A\bar{B}$			1	

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (a) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}C\bar{D} + \bar{A}BCD + ABCD + A\bar{B}CD + ABCD$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$				1
$\bar{A}B$		1	1	
AB		1	1	
$A\bar{B}$			1	

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped. (none here)*
7. Form the OR sum of terms generated by each loop.

Example (a) K-Map simplification

Simplify the following Boolean expression:

$\bar{A}\bar{B}\bar{C}D + \bar{A}BC\bar{D} + ABCD + ABCD + A\bar{B}CD + ABCD$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$				1
$\bar{A}B$		1	1	
AB		1	1	
$A\bar{B}$			1	

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

$BD + ACD + \bar{A}\bar{B}C\bar{D}$

Example (b) K-Map simplification

Simplify the following truth table:

A	B	C	D	X
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

Example (b) K-Map simplification

Simplify the following truth table:

A	B	C	D	X
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

Example (b) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}CD + \bar{A}\bar{B}C\bar{D} + \bar{A}BC\bar{D} + AB\bar{C}D + AB\bar{C}\bar{D} +$$

$$ABCD + ABCD$$

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (b) K-Map simplification

Simplify the following Boolean expression:

$\bar{A}\bar{B}CD + \bar{A}B\bar{C}\bar{D} + \bar{A}BC\bar{D} + AB\bar{C}D + AB\bar{C}\bar{D} +$

ABCD+ ABCD

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$			1	
$\bar{A}B$	1	1	1	1
AB	1	1		
$A\bar{B}$				

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (b) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}CD + \bar{A}B\bar{C}\bar{D} + \bar{A}BC\bar{D} + AB\bar{C}D + AB\bar{C}\bar{D} +$$

$$\bar{A}B\bar{C}D + \bar{A}BCD$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$			1	
$\bar{A}B$	1	1	1	1
AB	1	1		
$A\bar{B}$				

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s (none)
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (b) K-Map simplification

Simplify the following Boolean expression:

$\bar{A}\bar{B}CD + \bar{A}B\bar{C}\bar{D} + \bar{A}BC\bar{D} + AB\bar{C}D + AB\bar{C}\bar{D} +$

$ABCD + ABCD$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$			1	
$\bar{A}B$	1	1	1	1
AB	1	1		
$A\bar{B}$				

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (b) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}CD + \bar{A}B\bar{C}\bar{D} + \bar{A}BC\bar{D} + AB\bar{C}D + AB\bar{C}\bar{D} + \bar{A}B\bar{C}D + \bar{A}BCD$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$			1	
$\bar{A}B$	1	1	1	1
AB	1	1		
$A\bar{B}$				

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped. (none)*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (b) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}CD + \bar{A}B\bar{C}\bar{D} + \bar{A}B\bar{C}D + AB\bar{C}D + AB\bar{C}\bar{D} +$$

$$\bar{A}BCD + \bar{A}BC\bar{D}$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$			1	
$\bar{A}B$	1	1	1	1
AB	1	1		
$A\bar{B}$				

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Example (b) K-Map simplification

Simplify the following Boolean expression:

$$\bar{A}\bar{B}CD + A\bar{B}\bar{C}D + \bar{A}BC\bar{D} + AB\bar{C}D + AB\bar{C}\bar{D} +$$

$$ABCD + ABCD$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$			1	
$\bar{A}B$	1	1	1	1
AB	1	1		
$A\bar{B}$				

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped (none here)*
7. Form the OR sum of terms generated by each loop.

Example (b) K-Map simplification

Simplify the following Boolean expression:

$\bar{A}\bar{B}CD + \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}BC\bar{D} + AB\bar{C}D + AB\bar{C}\bar{D} +$

ABCD+ ABCD

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$			1	
$\bar{A}B$	1	1	1	1
AB	1	1		
$A\bar{B}$				

$B\bar{C} + \bar{A}CD + \bar{A}B$

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that have one or more 1s not already looped.
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

Try this one:

Simplify the following truth table using the K-Map method

5 minutes

A	B	C	D	X
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
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	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

Exercise: K-Map simplification

Boolean expression derived from the truth table:

$\bar{A}\bar{B}\bar{C}D + A\bar{B}CD + ABC\bar{D} + ABC\bar{D} + AB\bar{C}D + ABCD + \bar{A}BC$

$+ ABCD$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\ \bar{B}$		1		
$\bar{A}B$		1	1	1
AB	1	1	1	
$A\bar{B}$			1	

Exercise: K-Map simplification

Boolean expression derived from the truth table:

$\bar{A}\bar{B}\bar{C}D + \bar{A}B\bar{C}D + ABC\bar{D} + AB\bar{C}\bar{D} + A\bar{B}CD + ABCD + \bar{A}BC$

$+ ABCD$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$		1		
$\bar{A}B$		1	1	1
AB	1	1	1	
$A\bar{B}$			1	

1. Construct the K map, place 1s as per the truth table.
2. Loop 1s that are not adjacent to any other 1s.
3. Loop 1s that are in pairs *and cannot be looped into quads or octets.*
4. Loop 1s in octets (8) *even if they have already been looped.*
5. Loop quads (4) that *have one or more 1s not already looped.*
6. Loop any pairs (2) necessary to *include 1s not already looped.*
7. Form the OR sum of terms generated by each loop.

$\bar{A}\bar{C}D + \bar{A}BC + AB\bar{C} + ACD$

Don’t Care Output Conditions

Can be changed 0/1 so that the simplest expression can be obtained from the K-map. Typically occurs when we know certain input conditions are impossible.

Don't care Output Conditions

Can be changed 0/1 so that the simplest expression can be obtained from the K-map. Typically occur when we know certain input conditions are impossible.

A	B	C	z	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	x	"don't
1	0	0	x	care"
1	0	1	1	
1	1	0	1	
1	1	1	1	

(a)

Don’t care Output Conditions

Can be changed 0/1 so that the simplest expression can be obtained from the K-map. Typically occur when we know certain input conditions are impossible.

A	B	C	z	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	x	"don't care"
1	0	0	x	"don't care"
1	0	1	1	
1	1	0	1	
1	1	1	1	

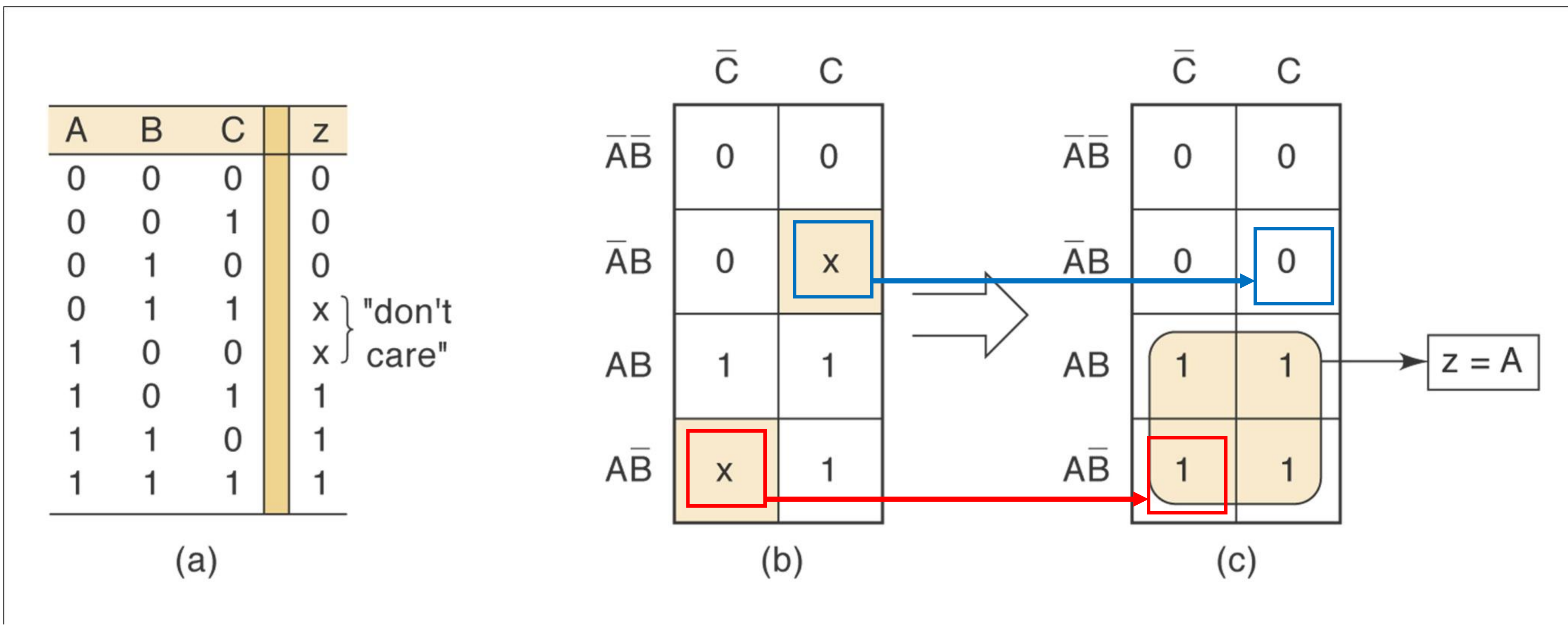
(a)

	$\overline{C}$	C
$\overline{A}\overline{B}$	0	0
$\overline{A}B$	0	x
AB	1	1
$A\overline{B}$	x	1

(b)

Don’t care Output Conditions

Can be changed 0/1 so that the simplest expression can be obtained from the K-map. Typically occur when we know certain input conditions are impossible.



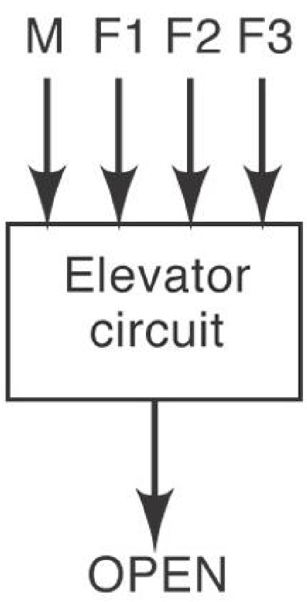
Example: Design a logic circuit for a three storey elevator.

Example: Design a logic circuit for a three-storey elevator.

M = Logic signal indicating if the elevator is moving (M = 1) or stationary (M = 0)

F1, F2 and F3 are the floor level signals, normally LO but go HI when a particular floor is reached.

The circuit output (O/P) is the "Door Open" signal, should be normally LO but go HI when the door is to open



M	F1	F2	F3		OPEN
0	0	0	0		
0	0	0	1		
0	0	1	0		
0	0	1	1		
0	1	0	0		
0	1	0	1		
0	1	1	0		
0	1	1	1		
1	0	0	0		
1	0	0	1		
1	0	1	0		
1	0	1	1		
1	1	0	0		
1	1	0	1		
1	1	1	0		
1	1	1	1		

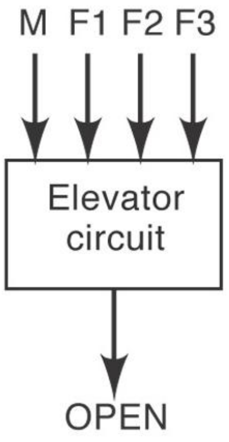
M = elevator moving

F1 = Floor 1

F2 – Floor 2

F3 – Floor 3

OPEN – elevator door opening



M	F1	F2	F3		OPEN
0	0	0	0		
0	0	0	1		
0	0	1	0		
0	0	1	1		
0	1	0	0		
0	1	0	1		
0	1	1	0		
0	1	1	1		
1	0	0	0		
1	0	0	1		
1	0	1	0		
1	0	1	1		
1	1	0	0		
1	1	0	1		
1	1	1	0		
1	1	1	1		

- Can only be on one floor at a time (only one floor I/P can be HI) .
- The other floor I/P's are then don't care conditions.
- Use x to indicate the don't care conditions.
- Door can't open when moving!

M	F1	F2	F3	OPEN
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	X
0	1	0	0	1
0	1	0	1	X
0	1	1	0	X
0	1	1	1	X
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	X
1	1	0	0	0
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

- Can only be on one floor at a time (only one floor I/P can be HI) .
- The other floor I/P's are then don't care conditions.
- Use x to indicate the don't care conditions.
- Door can't open when moving!

M	F1	F2	F3	OPEN
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	X
0	1	0	0	1
0	1	0	1	X
0	1	1	0	X
0	1	1	1	X
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	X
1	1	0	0	0
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

- Can only be on one floor at a time (only one floor I/P can be HI) .
- The other floor I/P's are then don't care conditions.
- Use x to indicate the don't care conditions.
- Door can't open when moving!

$$OPEN = \overline{M}\,\overline{F1}\,\overline{F2}\,F3 + \overline{M}\,F1\,\overline{F2}\,\overline{F3} + \overline{M}\,\overline{F1}\,F2\,\overline{F3}$$

	$\overline{F2}\,\overline{F3}$	$\overline{F2}\,F3$	$F2\,F3$	$F2\,\overline{F3}$
$\overline{M}\ \overline{F1}$	0	1	X	1
$\overline{M}\ F1$	1	X	X	X
$M\ F1$	0	X	X	X
$M\ \overline{F1}$	0	0	X	0

	$\overline{F2}\,\overline{F3}$	$\overline{F2}\,F3$	$F2\,F3$	$F2\,\overline{F3}$
$\overline{M}\ \overline{F1}$	0	1	X	1
$\overline{M}\ F1$	1	X	X	X
$M\ F1$	0	X	X	X
$M\ \overline{F1}$	0	0	X	0

	$\overline{F2}\,\overline{F3}$	$\overline{F2}\,F3$	$F2\,F3$	$F2\,\overline{F3}$
$\overline{M}\ \overline{F1}$	0	1	1	1
$\overline{M}\ F1$	1	1	1	1
$M\ F1$	0	0	0	0
$M\ \overline{F1}$	0	0	0	0

$$OPEN = \overline{M}\,(F1 + F2 + F3)$$

Exercises

Use the K-Map method to simplify the following:

a) AB + A(B + C) + B(B + C)

b) $A\overline{B} + A(\overline{B + C}) + B(\overline{B + C})$.

c) $[A\overline{B}(C + BD) + \overline{A}\,\overline{B}]C$

d) $\overline{A}BC + A\overline{B}\,\overline{C} + \overline{A}\,\overline{B}\,\overline{C} + A\overline{B}C + ABC$