ECED2200 – Digital Circuits

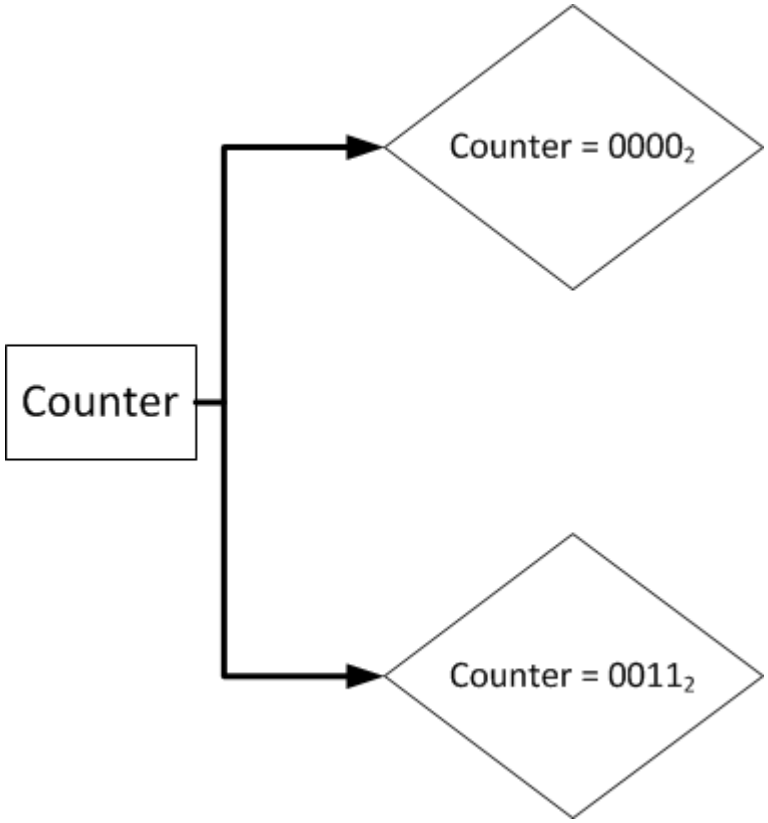
Karnaugh Maps

General Notes

- See updates to these slides: www.newae.com/teaching
- These slides licensed under 'Creative Commons Attribution-ShareAlike 3.0 Unported License'
- These slides are not the complete course – they are extended in-class
- You will find the following references useful, see www.newae.com/teaching for more information/links:
  - The book "Bebop to the Boolean Boogie" which is available to Dalhousie Students
  - Course notes (covers almost everything we will discuss in class)
  - Various websites such as e.g.: www.play-hookey.com
  - The book "Contemporary Logic Design", which was used in previous iterations of the class and you may have already

Gray Codes

Counters



1010

1001

1000

0111

0110

0101

0100

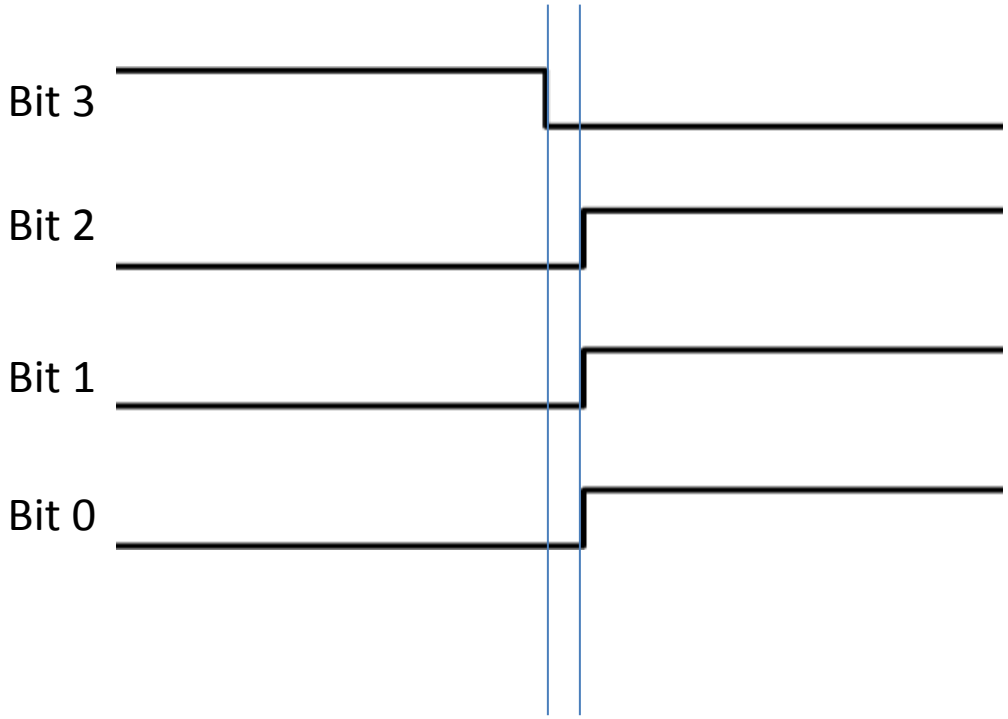
0011

0010

0001

0000

Oops...



What are Gray Codes?

0000

0001

0011

0010

0110

0111

0101

0100

1100

1101

1111

Generating Gray Codes

0

1

Generating Gray Codes

1.	2.	3.
0	0	00
1	1	01
	1	11
	0	10

2-bit Gray Code

1.	2.	3.
00	00	000
01	01	001
11	11	011
10	10	010
	10	110
	11	111
	01	101
	00	100

3-bit Gray Code

1. Write known Gray code down, even just 1-bit Gray Code
2. Mirror Gray code vertically
3. Add a single ‘0’ infront of original code, add a single ‘1’ infront of mirrored code
4. Repeat until required # of bits made

MINIMIZATION BY MAPPING

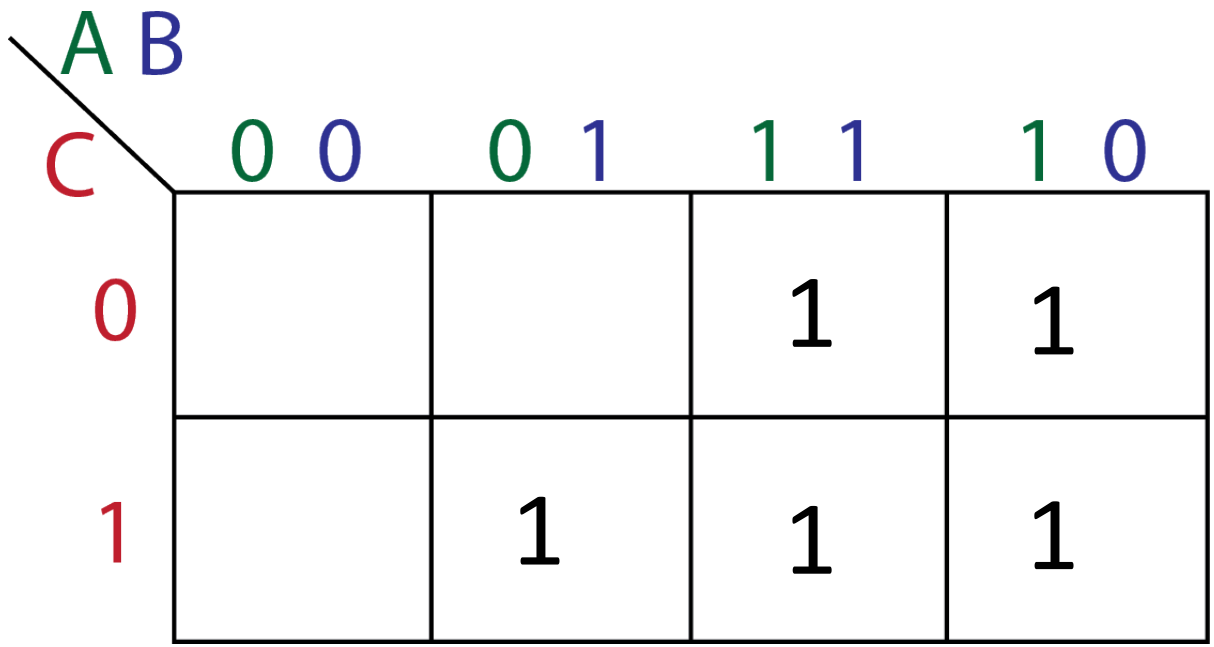
Karnaugh Maps

A	B	C	Minterms	f
0	0	0		0
0	0	1		0
0	1	0		0
0	1	1	$m_3 = \overline{A} \bullet B \bullet C$	1
1	0	0	$m_4 = A \bullet \overline{B} \bullet \overline{C}$	1
1	0	1	$m_5 = A \bullet \overline{B} \bullet C$	1
1	1	0	$m_6 = A \bullet B \bullet \overline{C}$	1
1	1	1	$m_7 = A \bullet B \bullet C$	1

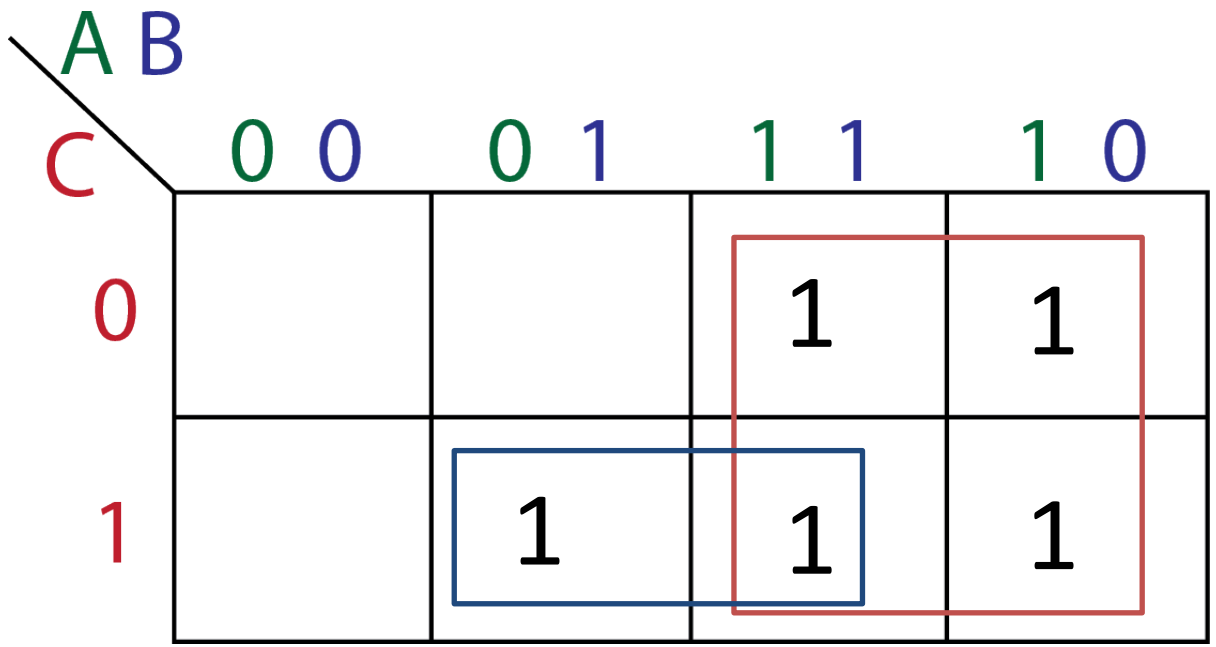
AB / C	00	01	11	10
0			1	1
1		1	1	1

$$f = m_3 + m_4 + m_5 + m_6 + m_7$$

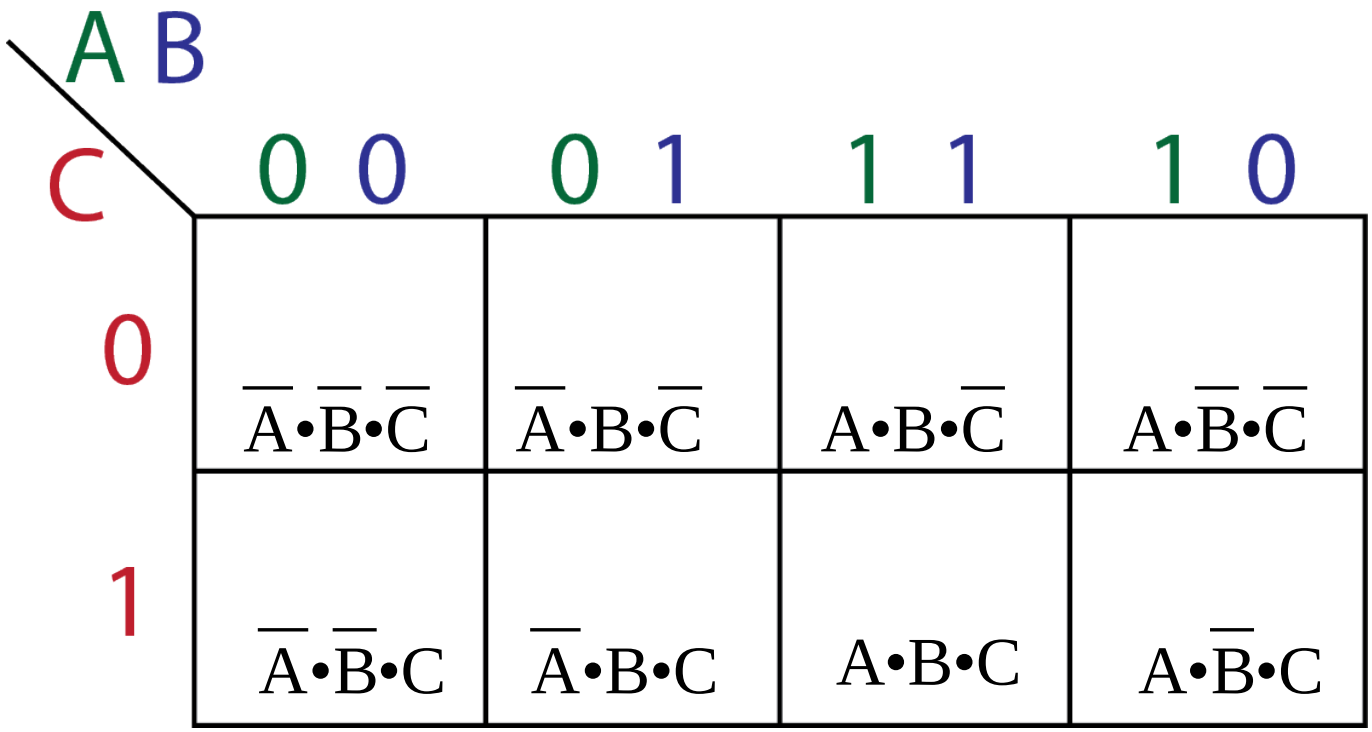
Karnaugh (K-Map) Minimization



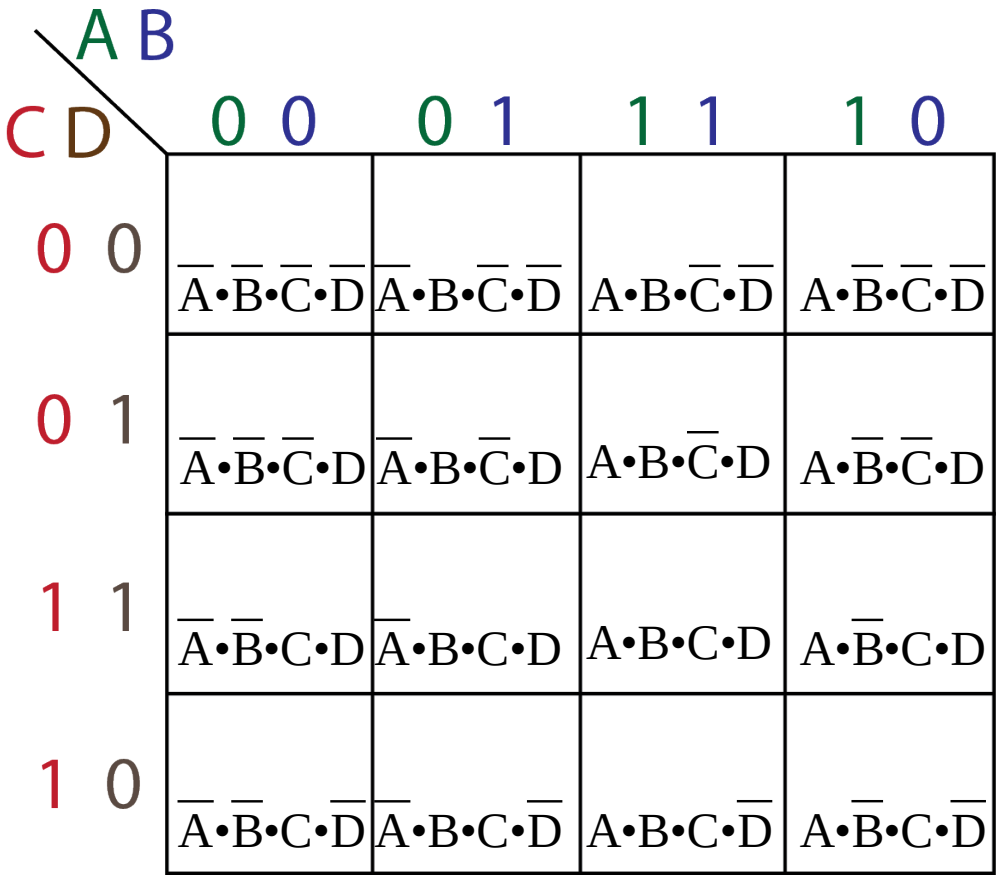
Karnaugh (K-Map) Minimization



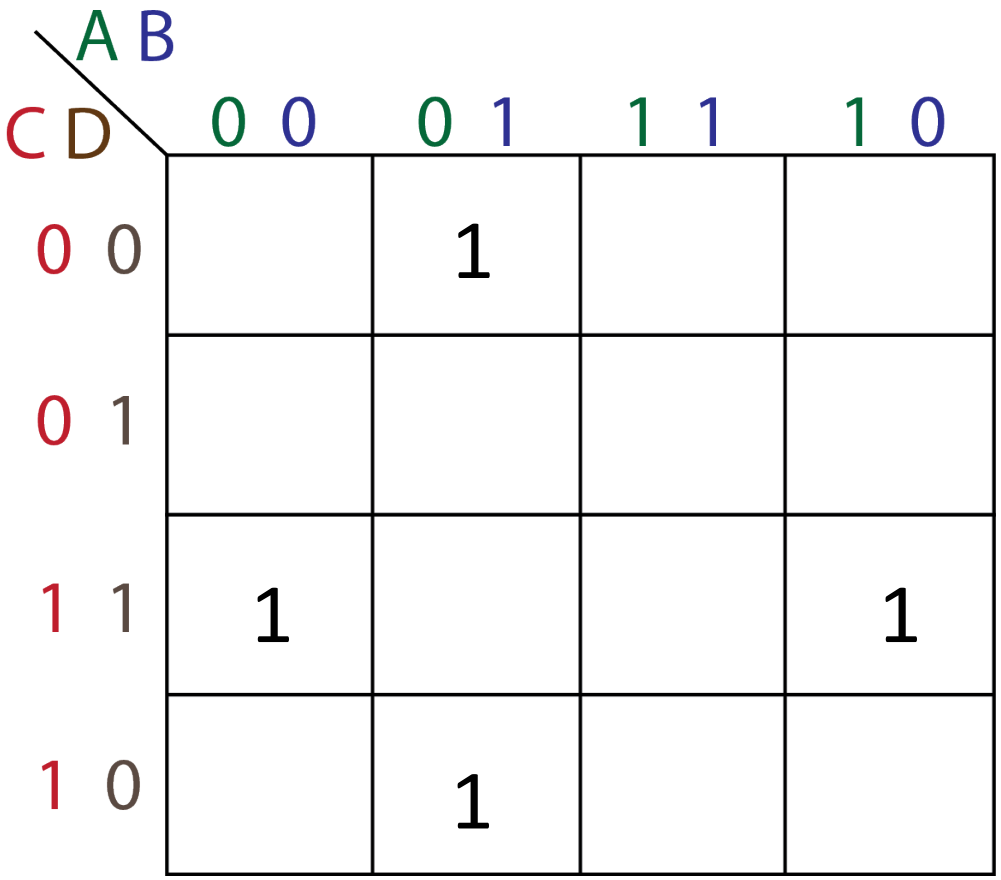
K-Map: 3-Input Product of Sum



K-Map: 4-Inputs Product of Sum

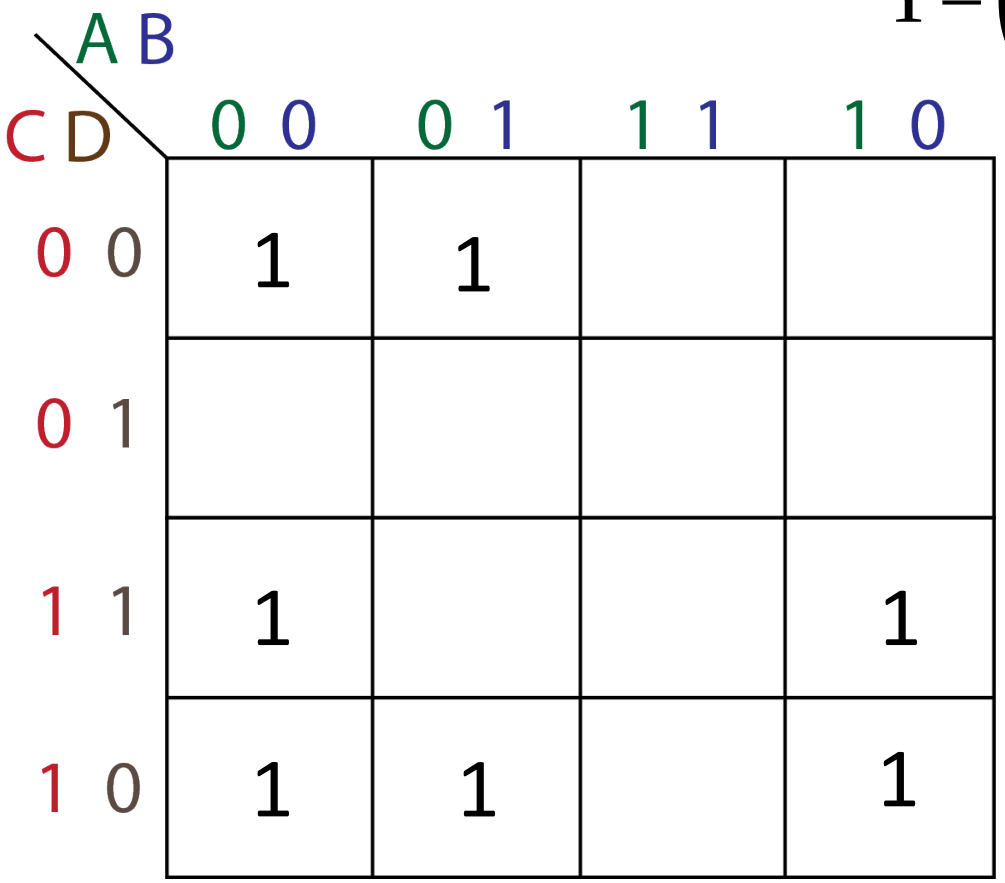


Mapping Connections

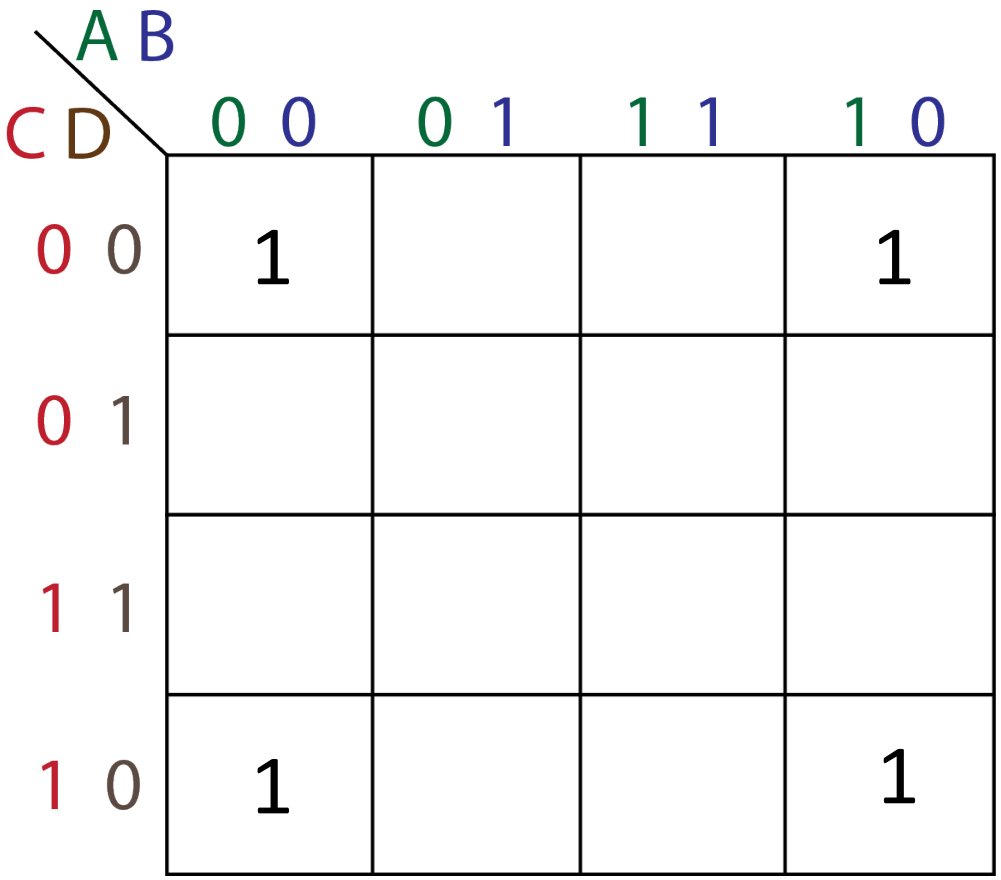


Mapping Connections

$$Y=\left(\overline{A}\bullet\overline{D}\right)+\left(\overline{B}\bullet C\right)$$

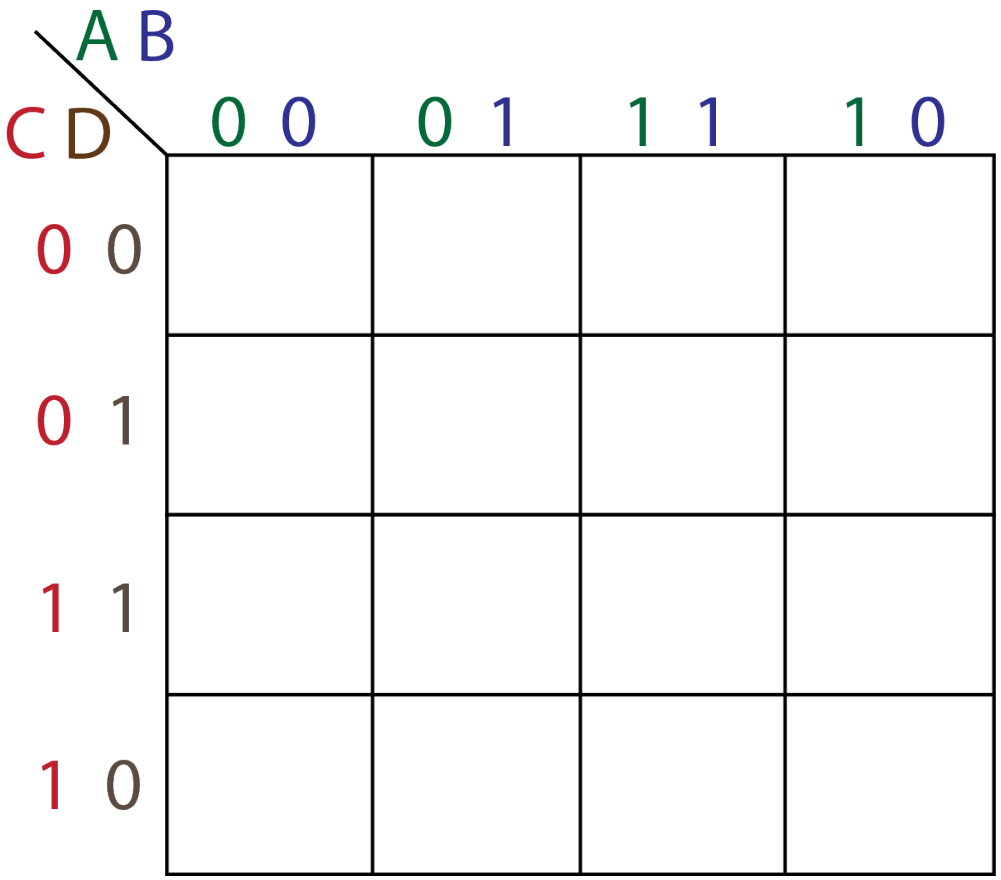


Mapping Connections



$$Y=\overline{B}\bullet\overline{D}$$

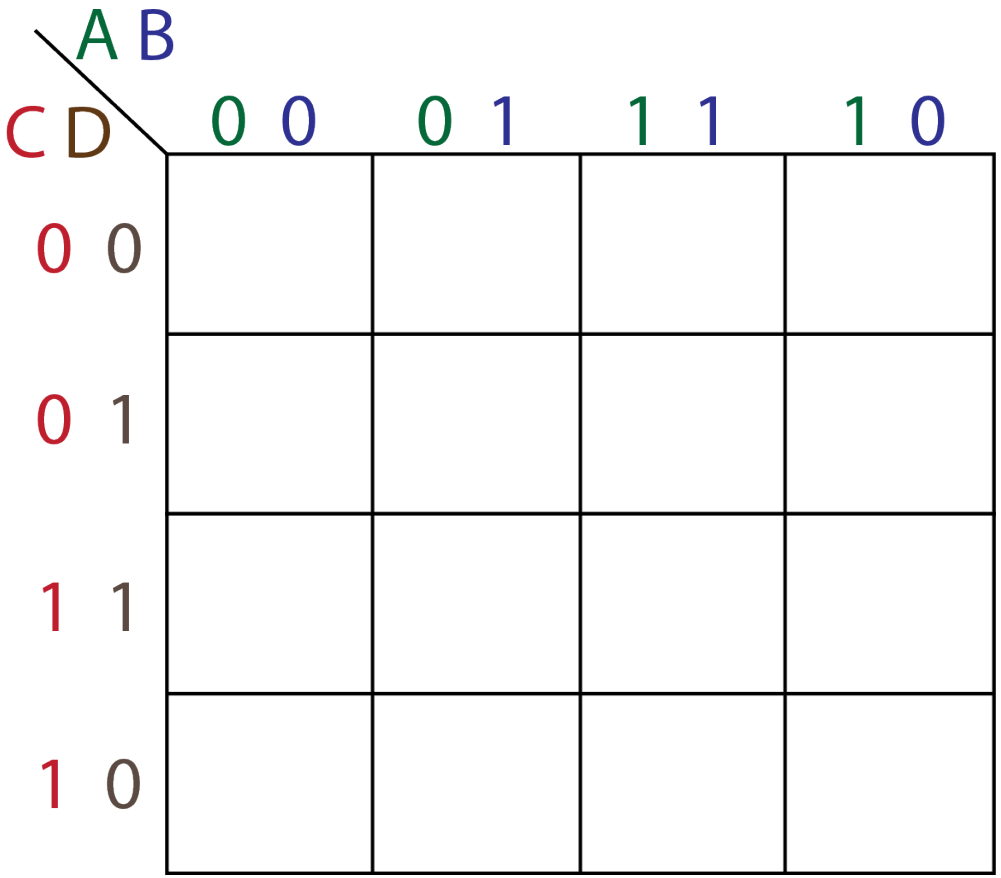
Mapping Connections



Notes on Grouping Terms

1. Groups must consist of 1,2,4,8,16,.. Etc terms
2. Left & Right-Side of K-Maps Connect
3. Top & Bottom of K-Maps Connect
4. Try to maximize number of terms per group
5. There may be multiple solutions to same problem
6. Based on if more 1's or 0's in truth table can use Product-of-Sum or Sum-of-Product resp.

Multiple Solutions



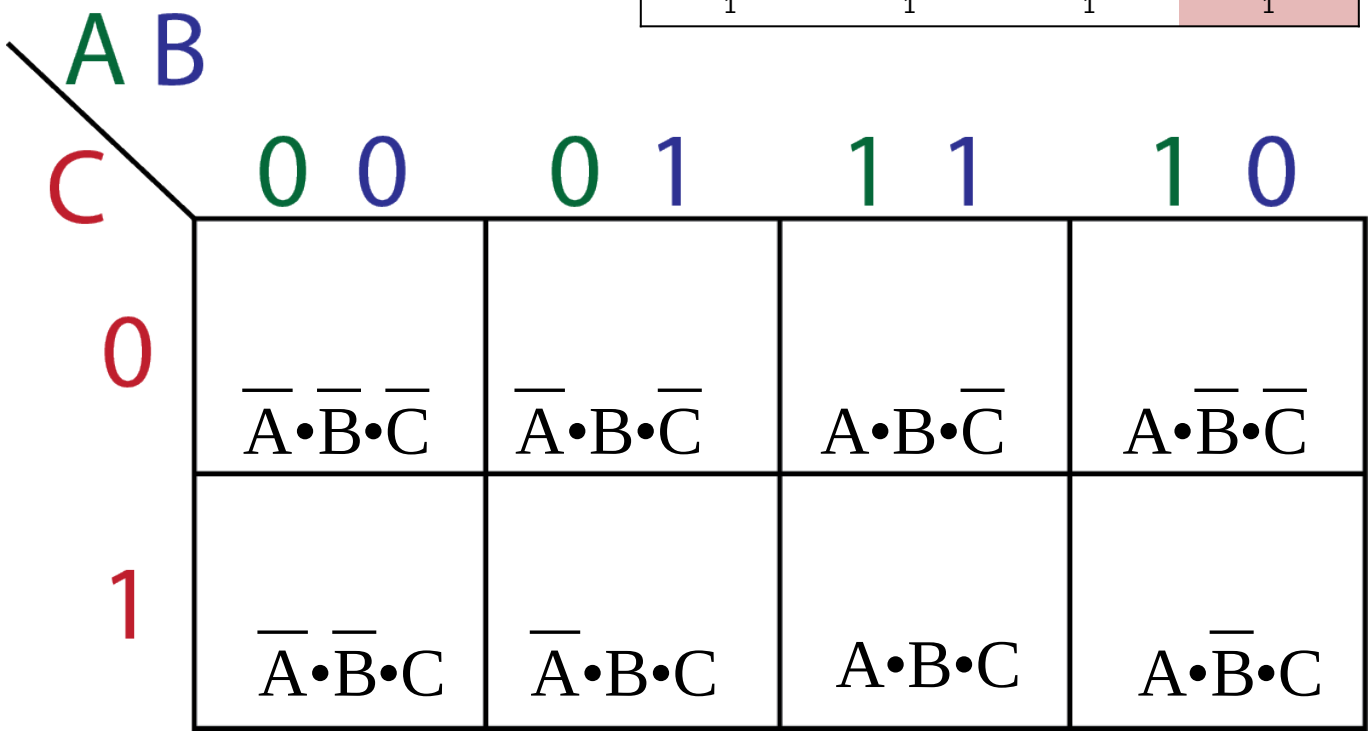
Example #1 – Vote Taker

Map the following:

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Example #1

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

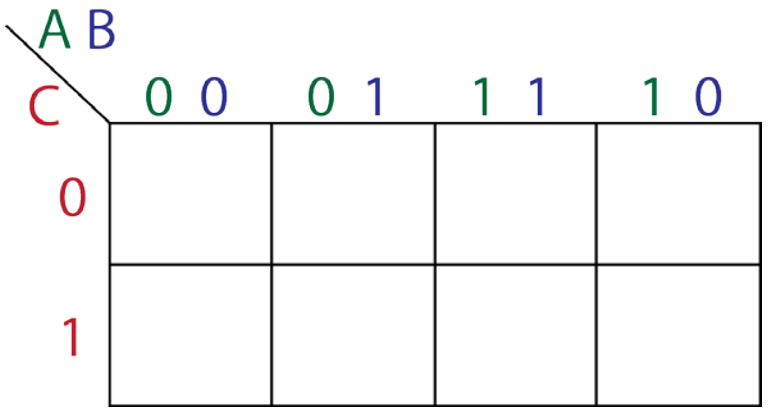


Example #2 – Full Adder

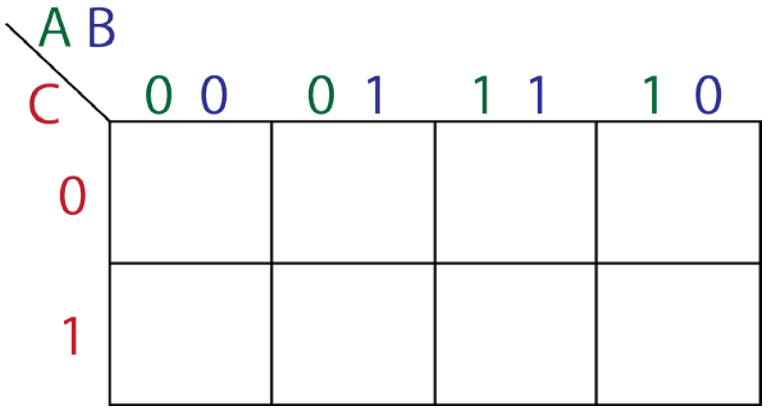
A	B	Cin	Sum	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Example #2 – Full Adder

A	B	Cin	Sum
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1



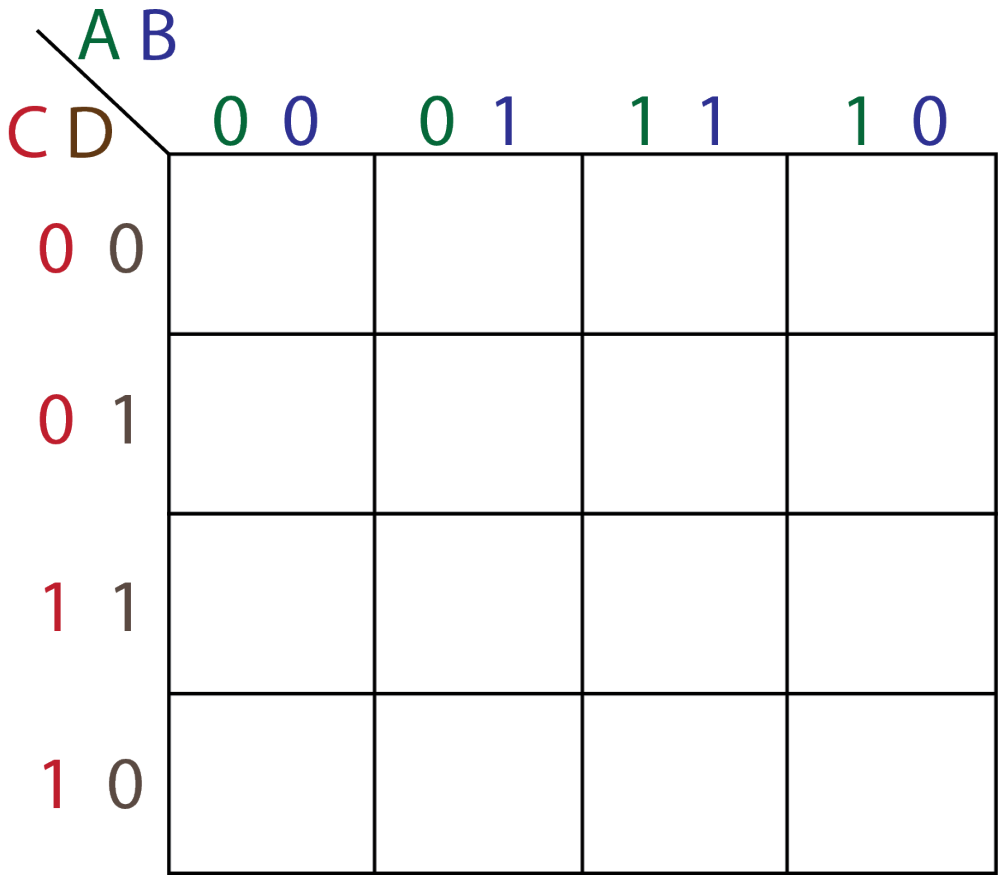
A	B	Cin	Cout
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Example #3 – Comparator

A	B	C	D	AB > CD
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Example #3 - Comparator

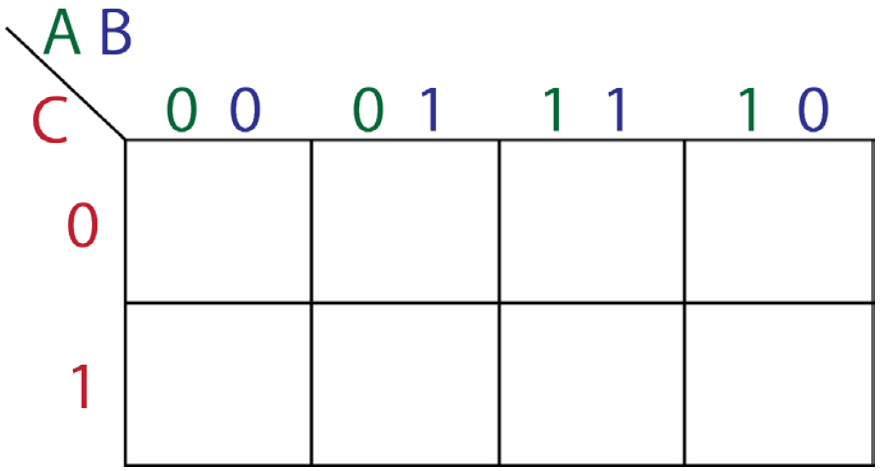


Mapping with Don't Cares

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	?
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	?

Mapping with Don't Cares

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	?
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	?



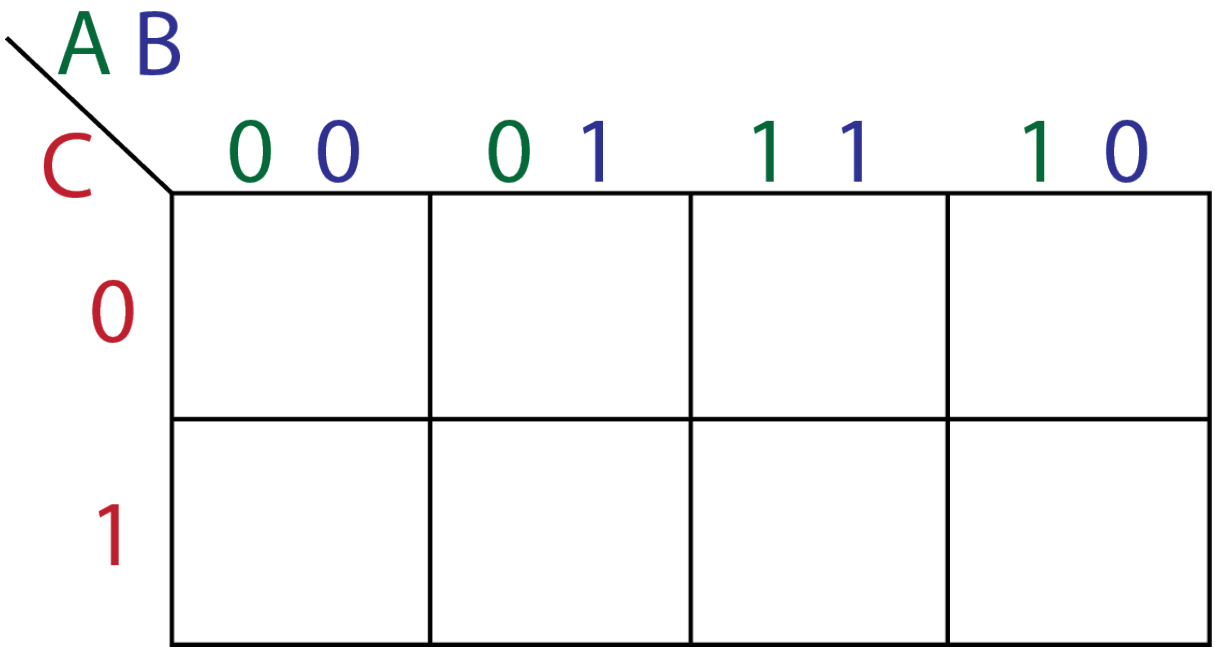
Note on Don't Care Notation

Product of Sum Mapping

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

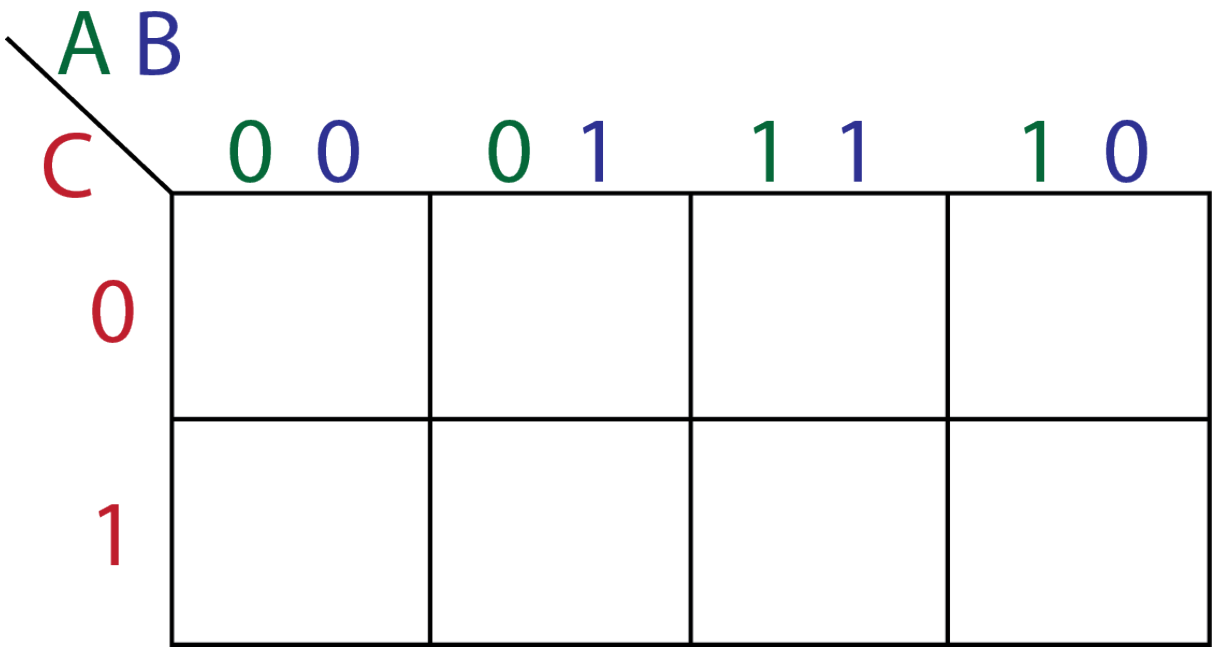
Mapping using SoP

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

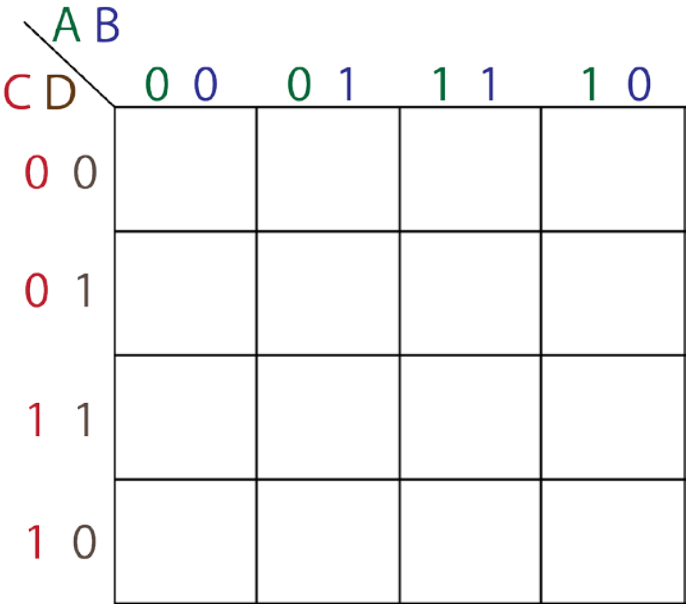


Mapping using PoS

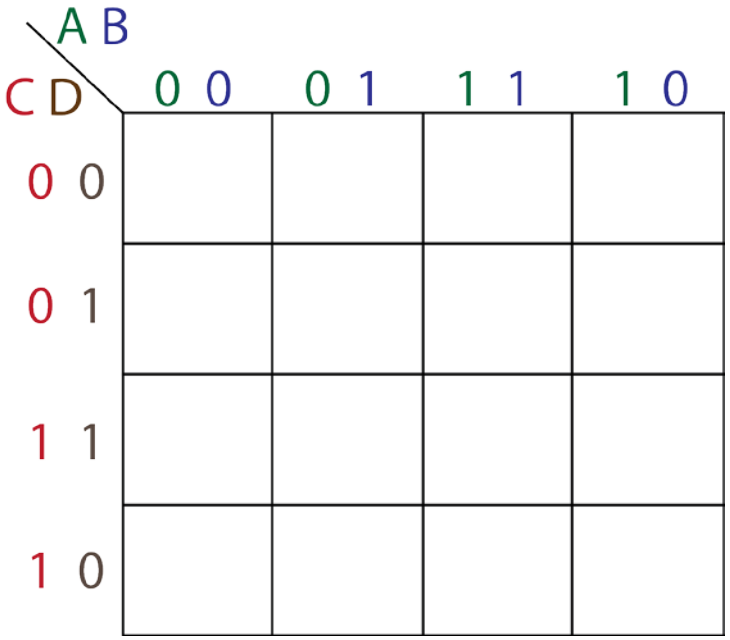
A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Mapping 5 Input Variables

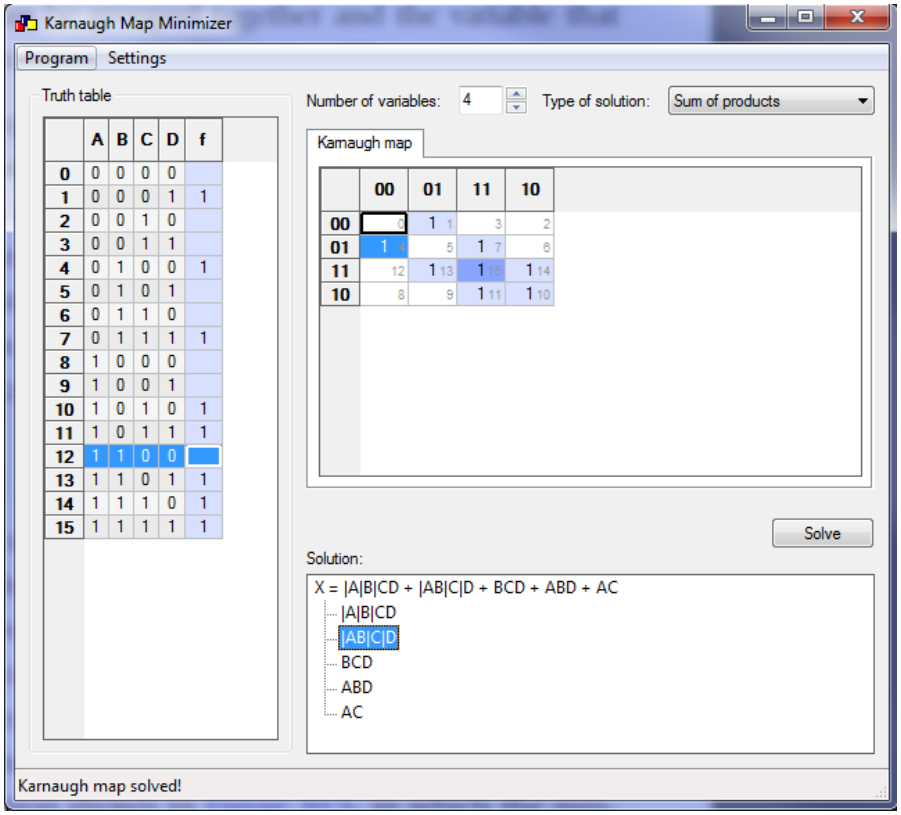


E=0



E=1

Checking your Results



http://k-map.sourceforge.net

Section Summary

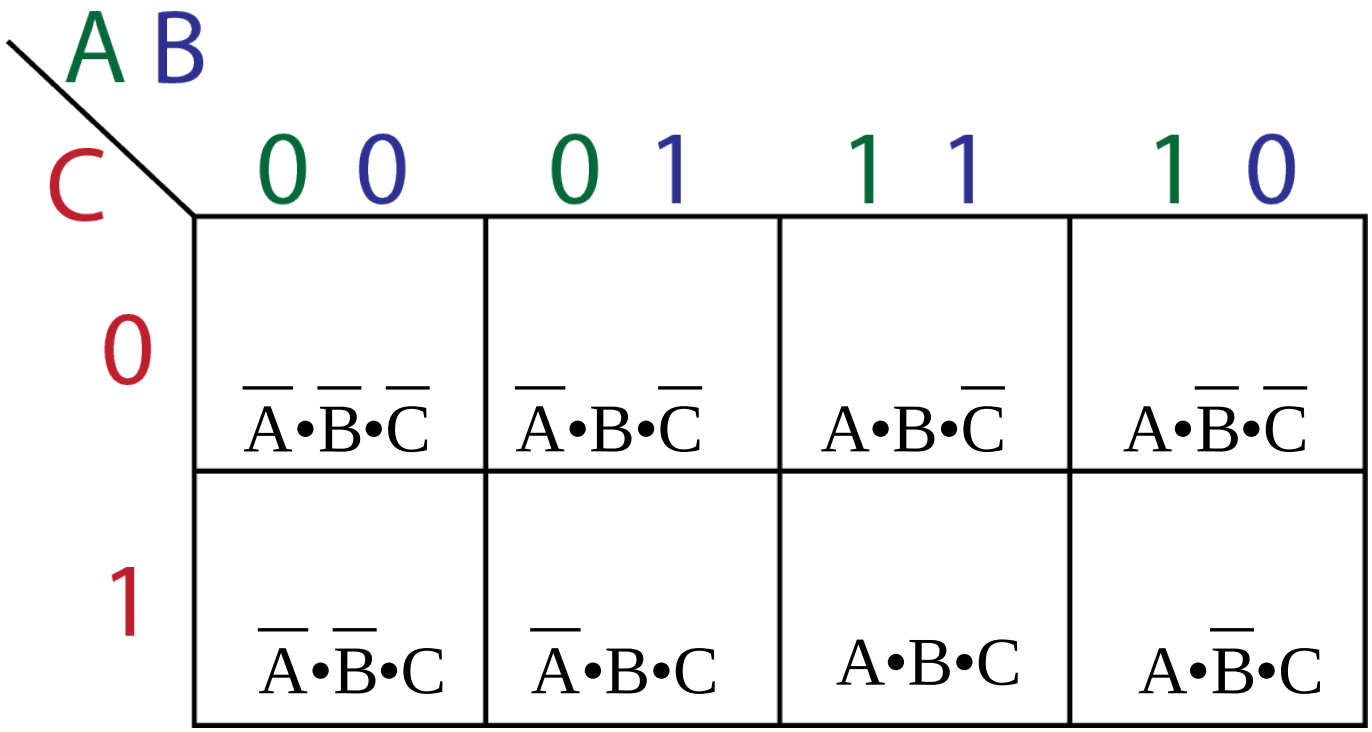
- See ECED2200 Notes “Minimization By Mapping” (Page 76)
- Bebop to the Boolean Boogie Chapter 10
- Contemporary Logic Design Chapter 2

Useful software:

http://k-map.sourceforge.net/

MAPPING EQUATIONS

$$A \cdot B \cdot C \; + \; A \cdot B \cdot \overline{C} \; + \; A \cdot \overline{B} \cdot \overline{C}$$



$$A \cdot B \cdot C \cdot \overline{D} + A \cdot \overline{B} \cdot C + \overline{B} \cdot \overline{C}$$

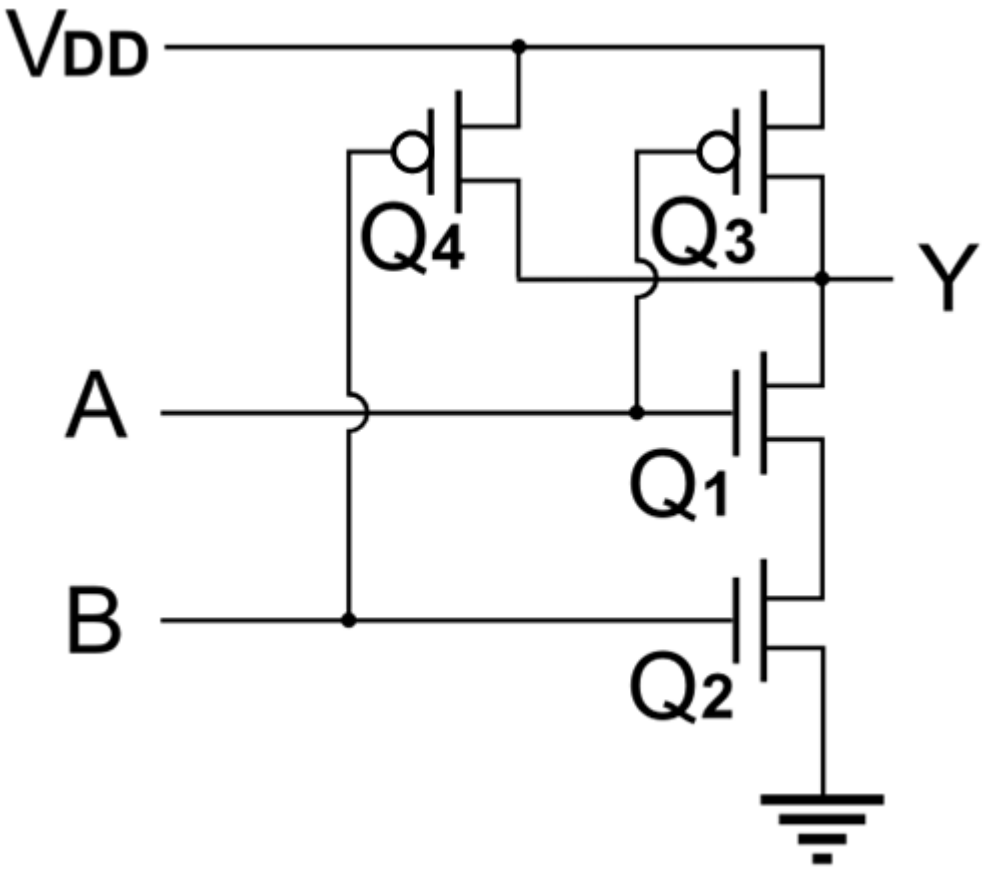
C D \ A B	0 0	0 1	1 1	1 0
0 0	$\overline{A}\bullet\overline{B}\bullet\overline{C}\bullet\overline{D}$	$\overline{A}\bullet B\bullet\overline{C}\bullet\overline{D}$	$A\bullet B\bullet\overline{C}\bullet\overline{D}$	$A\bullet\overline{B}\bullet\overline{C}\bullet\overline{D}$
0 1	$\overline{A}\bullet\overline{B}\bullet\overline{C}\bullet D$	$\overline{A}\bullet B\bullet\overline{C}\bullet D$	$A\bullet B\bullet\overline{C}\bullet D$	$A\bullet\overline{B}\bullet\overline{C}\bullet D$
1 1	$\overline{A}\bullet\overline{B}\bullet C\bullet D$	$\overline{A}\bullet B\bullet C\bullet D$	$A\bullet B\bullet C\bullet D$	$A\bullet\overline{B}\bullet C\bullet D$
1 0	$\overline{A}\bullet\overline{B}\bullet C\bullet\overline{D}$	$\overline{A}\bullet B\bullet C\bullet\overline{D}$	$A\bullet B\bullet C\bullet\overline{D}$	$A\bullet\overline{B}\bullet C\bullet\overline{D}$

Section Summary

- See ECED2200 Notes “Minimization By Mapping” (Page 76)
- Bebop to the Boolean Boogie Chapter 10
- Contemporary Logic Design Chapter 2

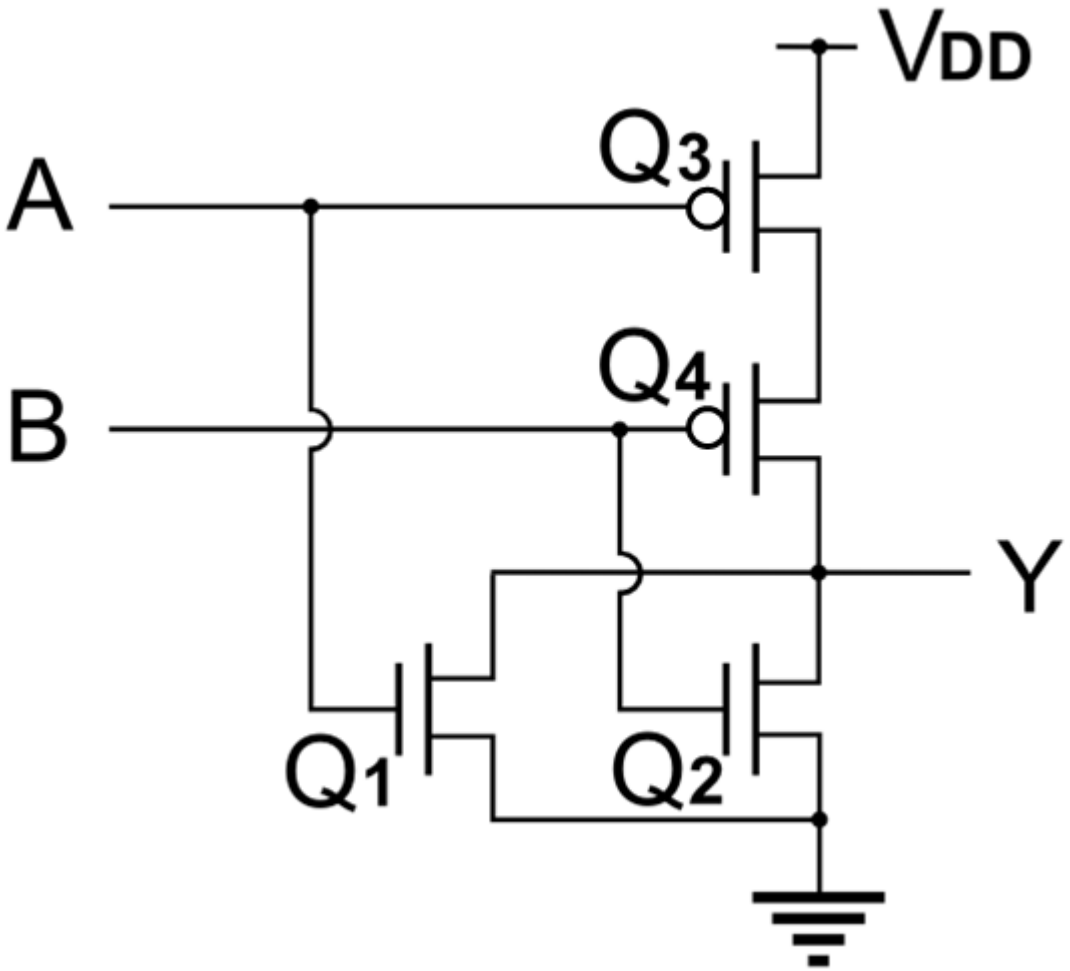
NAND – NOR CONVERSIONS

FET Logic Gates - NAND



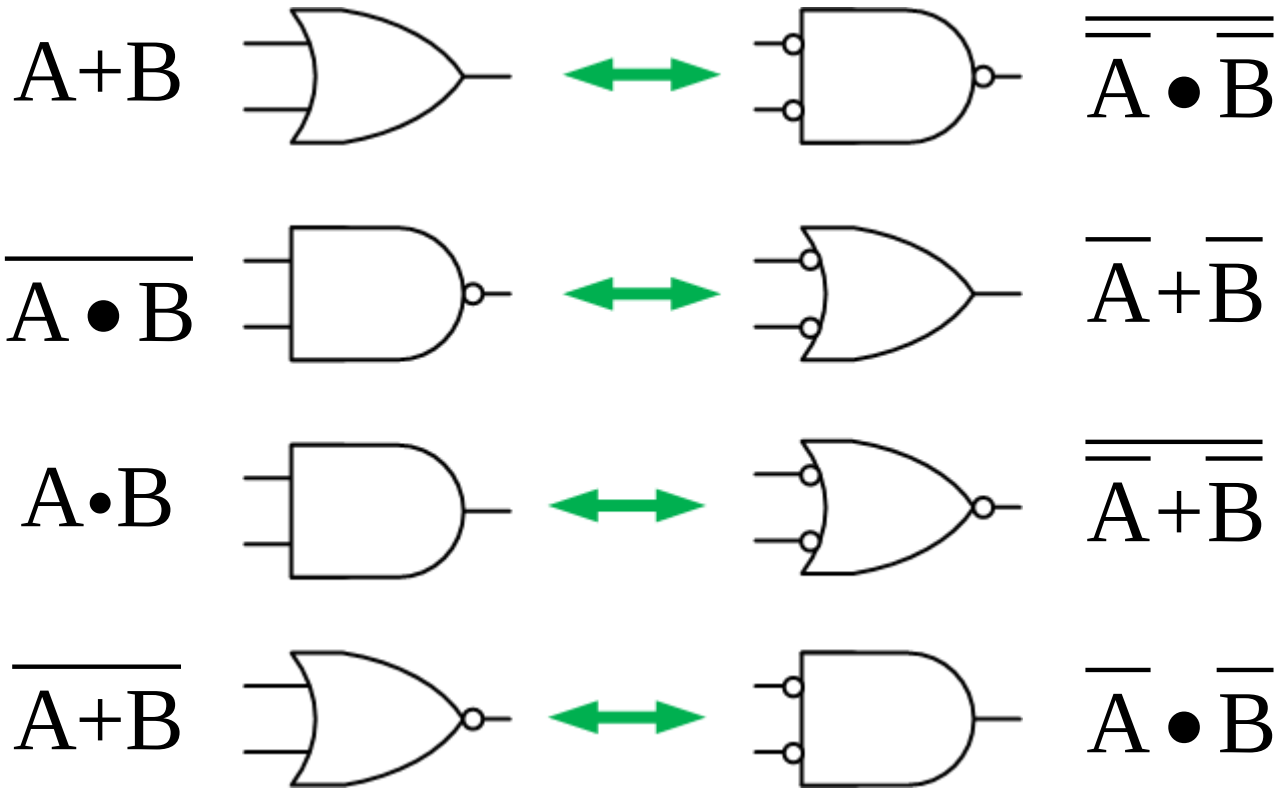
Source: http://commons.wikimedia.org/wiki/File:NAND_gate_(CMOS_circuit).PNG

FET Logic Gates - NOR



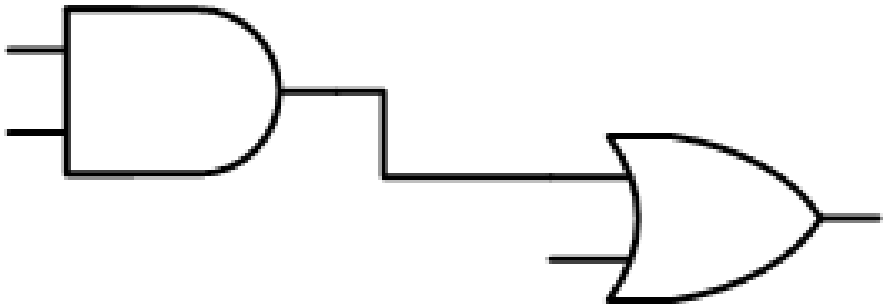
http://commons.wikimedia.org/wiki/File:NOR_gate_%28CMOS_circuit%29.PNG

Equivalencies



NAND Conversion: Step 1

1. Draw complete schematic

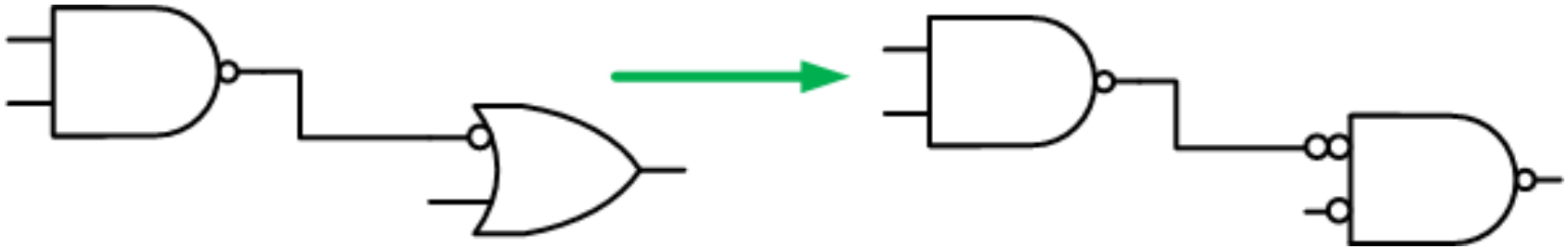


NAND Conversion: Step 2

Convert all AND gates to NAND gates and add additional inverted input to next level:

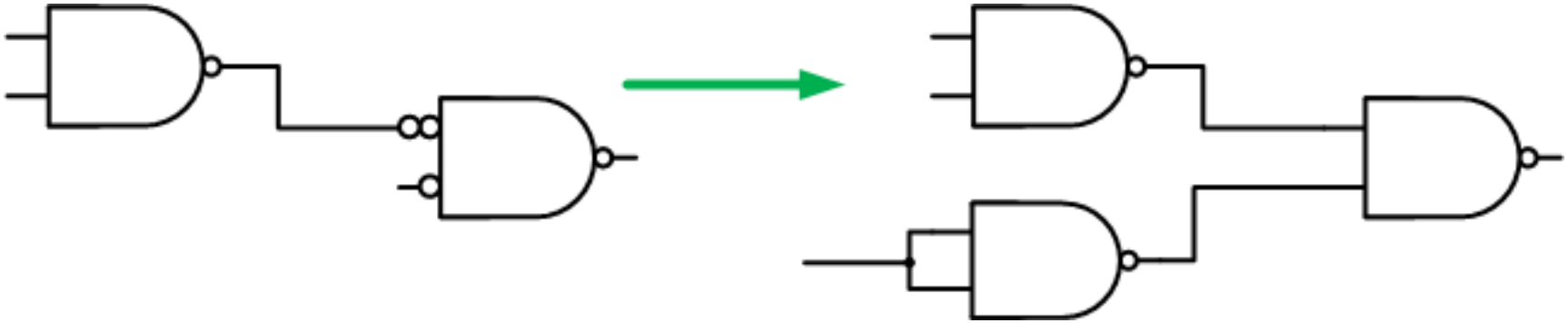
NAND Conversion: Step 3

Convert all OR gates to NAND gates:



NAND Conversion: Step 4

If two circles connect together cancel them. If circles do not cancel on inputs to NAND gates, add inverter built from NAND.



Section Summary

- See ECED2200 Notes “Conversion to NAND and NOR Networks” (Page 102)