CPSC 359 – Digital Logic Tutorial #1

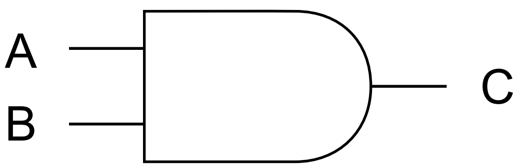
Basic Gates

Andrew Kuipers

Basic Gates

AND

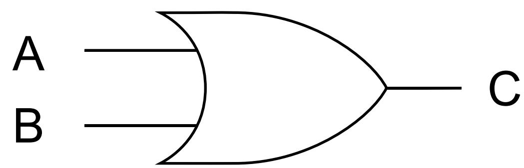
$$C = A \wedge B$$



A	B	A ∧ B
0	0	0
0	1	0
1	0	0
1	1	1

OR

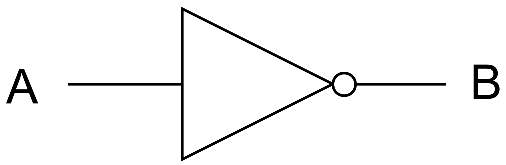
$$C = A \vee B$$



A	B	A ∨ B
0	0	0
0	1	1
1	0	1
1	1	1

NOT

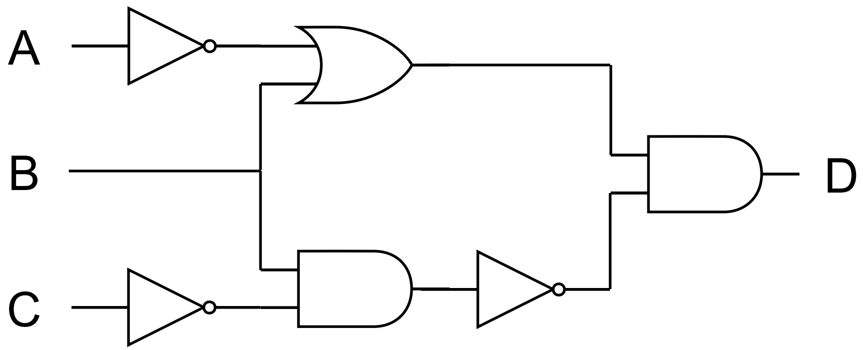
$$B = \neg A$$



A	¬A
0	1
1	0

Example

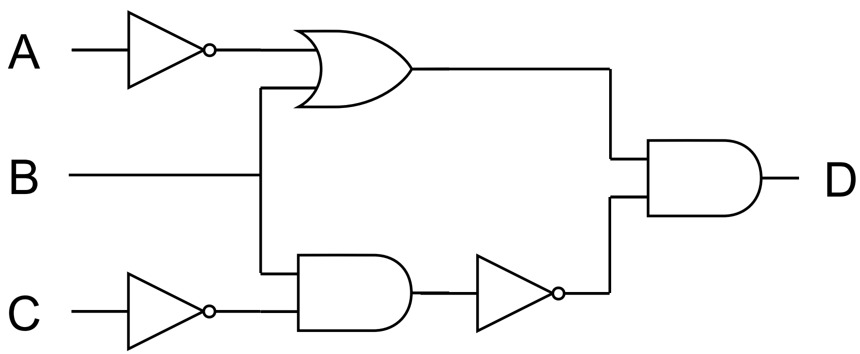
$$D = (\neg A \vee B) \wedge \neg(B \wedge \neg C)$$



A	B	C	$(\neg A \vee B) \wedge \neg(B \wedge \neg C)$					
0	0	0						
0	0	1						
0	1	0						
0	1	1						
1	0	0						
1	0	1						
1	1	0						
1	1	1						

Example

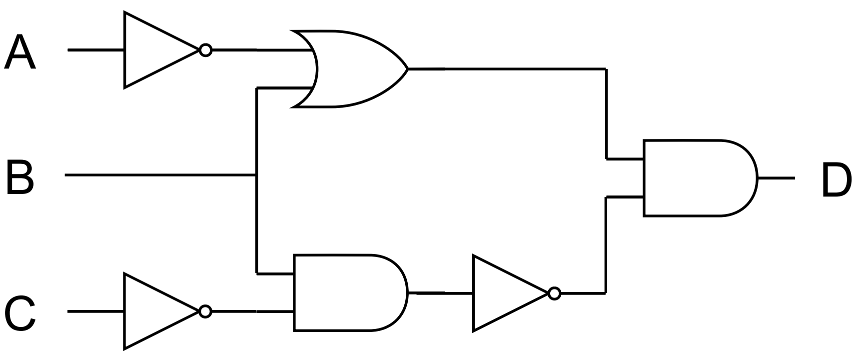
$D = (\neg A \vee B) \wedge \neg(B \wedge \neg C)$



A	B	C	$(\neg A \vee B) \wedge \neg(B \wedge \neg C)$					
0	0	0						1
0	0	1						0
0	1	0						1
0	1	1						0
1	0	0						1
1	0	1						0
1	1	0						1
1	1	1						0

Example

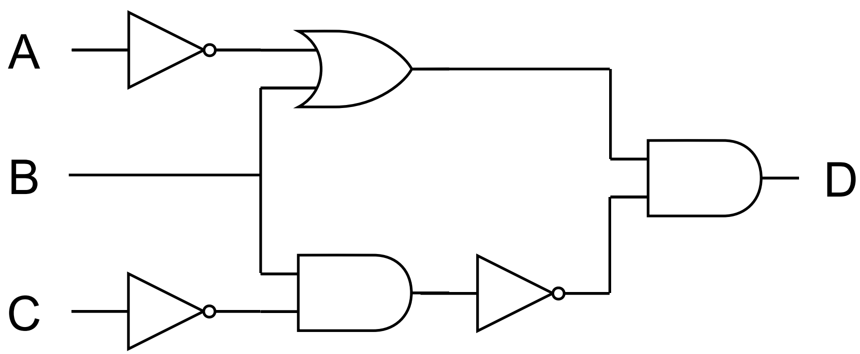
$$D = (\neg A \vee B) \wedge \neg(B \wedge \neg C)$$



A	B	C	$(\neg A \vee B) \wedge \neg(B \wedge \neg C)$					
0	0	0					0	1
0	0	1					0	0
0	1	0					1	1
0	1	1					0	0
1	0	0					0	1
1	0	1					0	0
1	1	0					1	1
1	1	1					0	0

Example

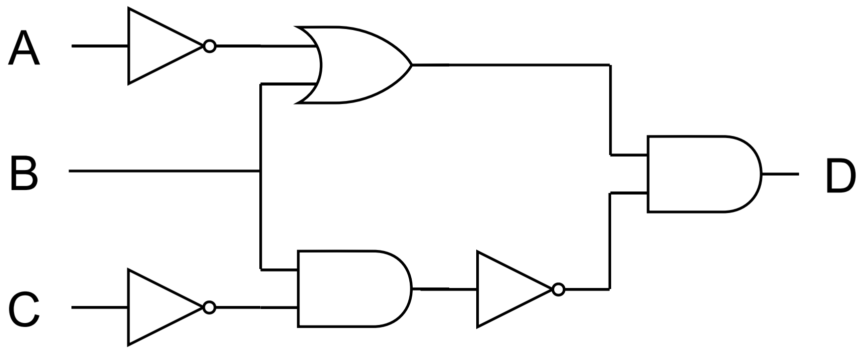
$$D = (\neg A \lor B) \land \neg(B \land \neg C)$$



A	B	C	$(\neg A \lor B) \land \neg(B \land \neg C)$					
0	0	0				1	0	1
0	0	1				1	0	0
0	1	0				0	1	1
0	1	1				1	0	0
1	0	0				1	0	1
1	0	1				1	0	0
1	1	0				0	1	1
1	1	1				1	0	0

Example

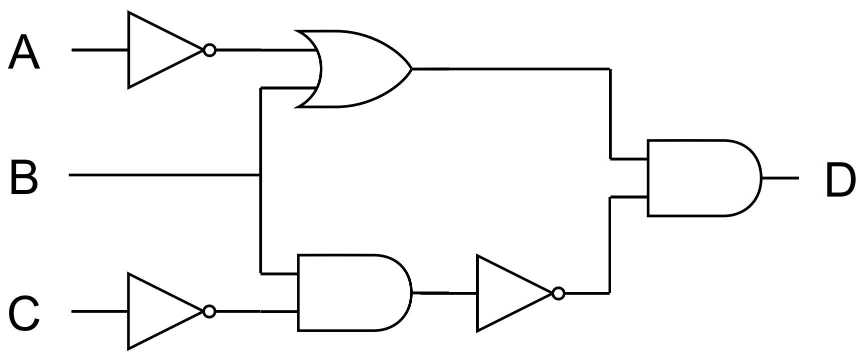
$$D = (\neg A \vee B) \wedge \neg(B \wedge \neg C)$$



A	B	C	$(\neg A \vee B) \wedge \neg(B \wedge \neg C)$					
0	0	0	1			1	0	1
0	0	1	1			1	0	0
0	1	0	1			0	1	1
0	1	1	1			1	0	0
1	0	0	0			1	0	1
1	0	1	0			1	0	0
1	1	0	0			0	1	1
1	1	1	0			1	0	0

Example

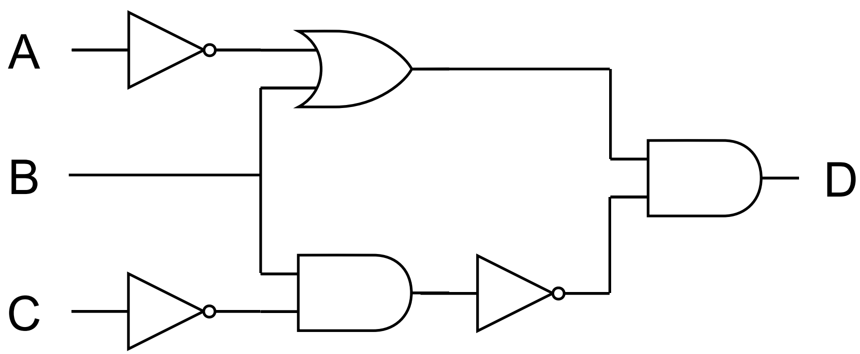
$$D = (\neg A \vee B) \wedge \neg(B \wedge \neg C)$$



A	B	C	$(\neg A \vee B) \wedge \neg(B \wedge \neg C)$					
0	0	0	1	1		1	0	1
0	0	1	1	1		1	0	0
0	1	0	1	1		0	1	1
0	1	1	1	1		1	0	0
1	0	0	0	0		1	0	1
1	0	1	0	0		1	0	0
1	1	0	0	1		0	1	1
1	1	1	0	1		1	0	0

Example

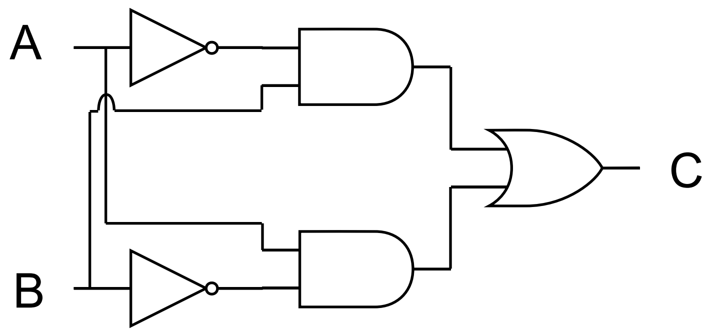
$$D = (\neg A \vee B) \wedge \neg(B \wedge \neg C)$$



A	B	C	$(\neg A \vee B) \wedge \neg(B \wedge \neg C)$					
0	0	0	1	1	1	1	0	1
0	0	1	1	1	1	1	0	0
0	1	0	1	1	0	0	1	1
0	1	1	1	1	1	1	0	0
1	0	0	0	0	0	1	0	1
1	0	1	0	0	0	1	0	0
1	1	0	0	1	0	0	1	1
1	1	1	0	1	1	1	0	0

Exercise

1. Translate to a logical formula:



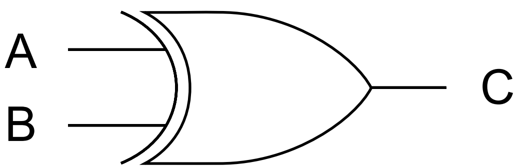
2. Create a circuit for the logical formula:

$$D = \neg(A \wedge \neg B) \wedge (C \vee \neg A \vee B)$$

Other Gates

XOR

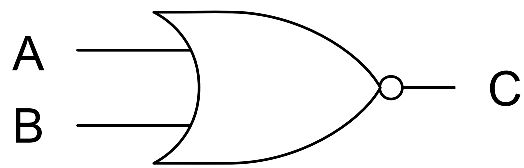
$$C = A \oplus B$$



A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

NOR

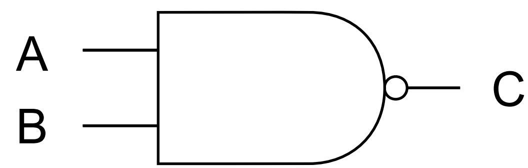
$$C = \neg(A \vee B)$$



A	B	$\neg(A \vee B)$
0	0	1
0	1	0
1	0	0
1	1	0

NAND

$$C = \neg(A \wedge B)$$



A	B	$\neg(A \wedge B)$
0	0	1
0	1	1
1	0	1
1	1	0

Transforming Circuits

De Morgan's Laws

$\neg(A \wedge B) \Leftrightarrow \neg A \vee \neg B$

$\neg(A \vee B) \Leftrightarrow \neg A \wedge \neg B$

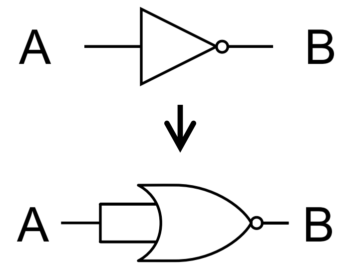
So: $A \wedge B \Leftrightarrow \neg(\neg A \vee \neg B)$ *and* $A \vee B \Leftrightarrow \neg(\neg A \wedge \neg B)$

Therefore: Just need $\{ \wedge, \neg \}$ or $\{ \vee, \neg \}$ for any circuit

Transforming Circuits

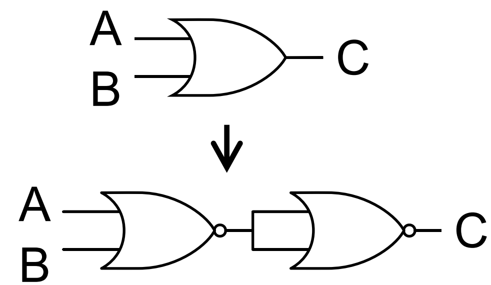
NOR can make a NOT gate:

$$\neg A \Leftrightarrow \neg A \wedge \neg A \Leftrightarrow \neg(A \vee A)$$



NOR can make an OR gate:

$$A \vee B \Leftrightarrow \neg(\neg(A \vee B))$$

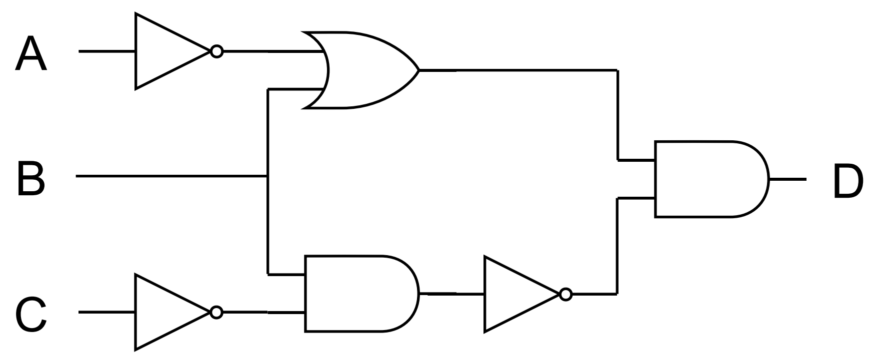


So: NOR gates alone can be used to make any circuit

* can NAND gates do the same thing?

Exercise

Translate into a circuit using only NOR gates:



* Can you optimize the number of gates used?