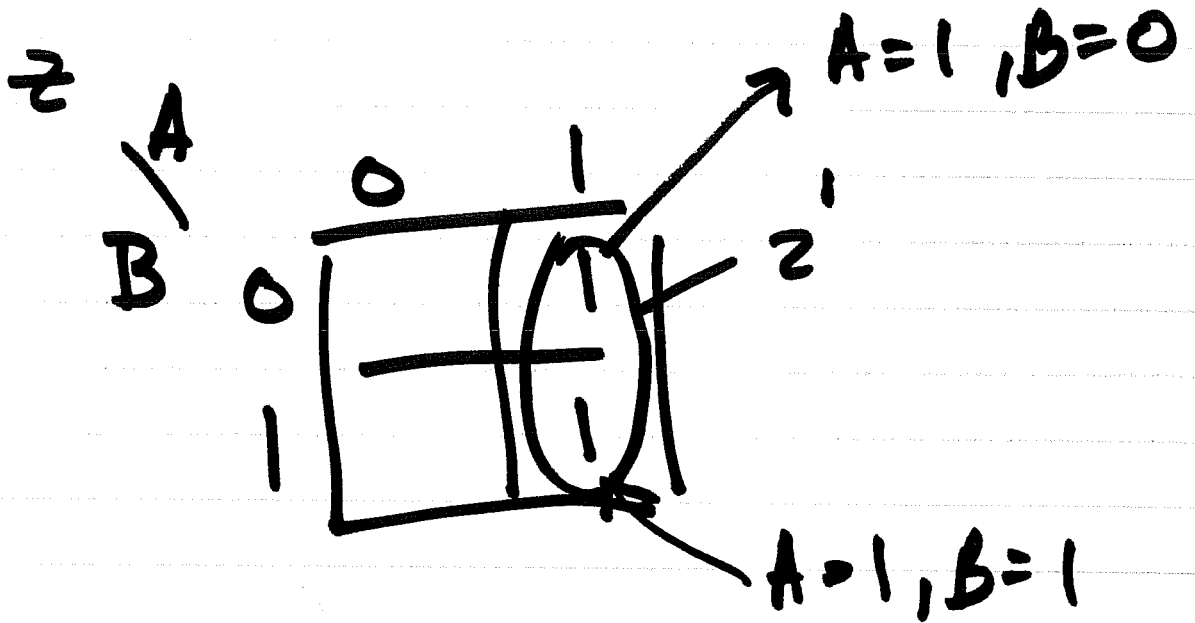
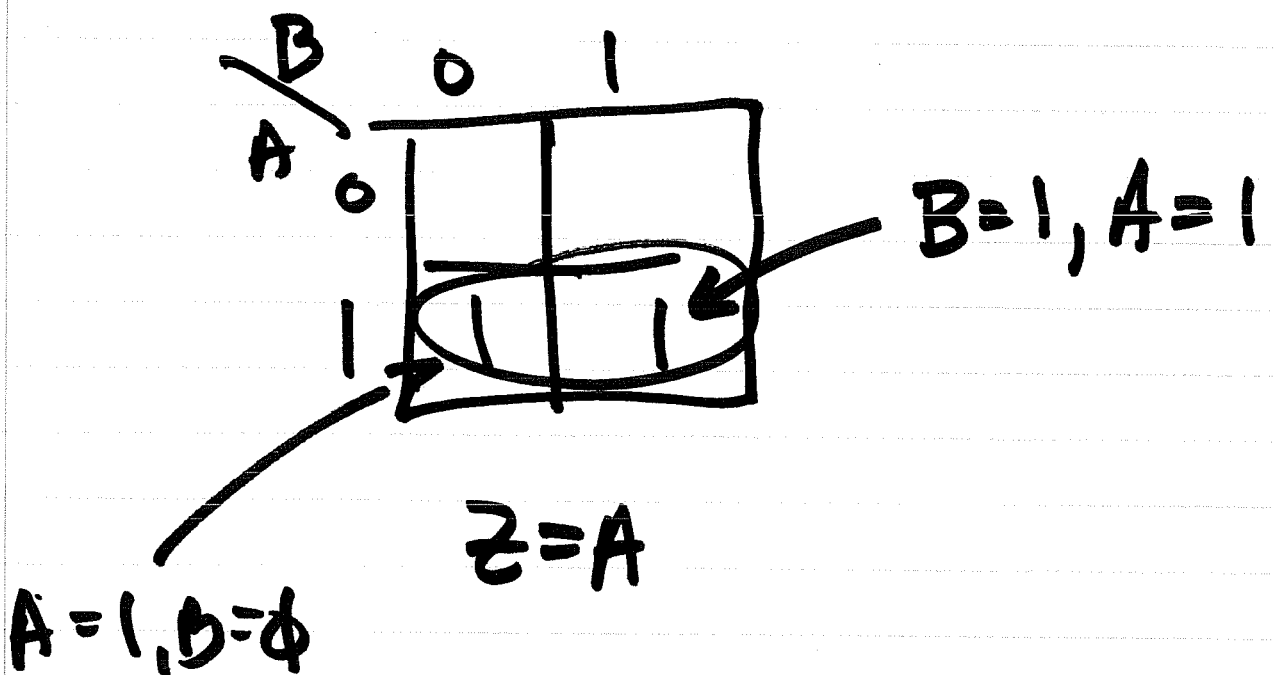


K-maps



$z = A$



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

$$Y = C \cdot + AB$$

Prime
Implicants (PI)

		AB			
		00	01	11	10
CD	00			1	1
	01	1	1		
	11	1	1		
	10				1

$$Z = B \cdot D + A \bar{B} \bar{D} + A \bar{C} \bar{D}$$

$$= B \cdot D + A \bar{D} (\bar{B} + \bar{C})$$

Quine - McCluskey Technique
(Tabular Method)

Heuristic Method -

↳ espresso

↳ BLIF

Berkeley Logic Interchange

Format
ASCII → binary-text

.i	3	←	# inputs
.o	1	←	# outputs
.ilb	a b c	←	names inputs
.ob	z	←	names outputs

space

000	1
001	0
010	1
011	0
100	1
111	0

← end of file

espresso ^ -Dexact ^ -oegntott ^)
↑
space
↑
op
→ file.txt

! = NOT

& = AND

| = OR

→ Schematic

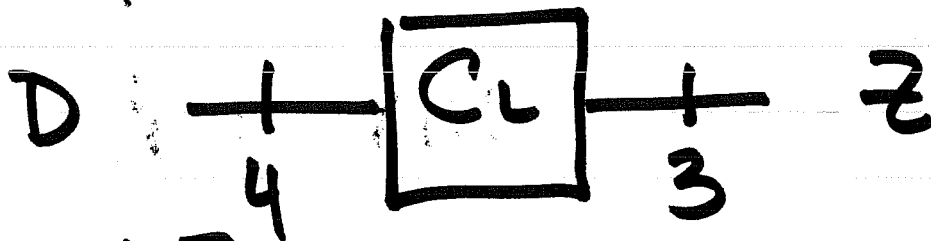
Boolean equation

→ Hardware Descriptive Language (HDL)

[Verilog]

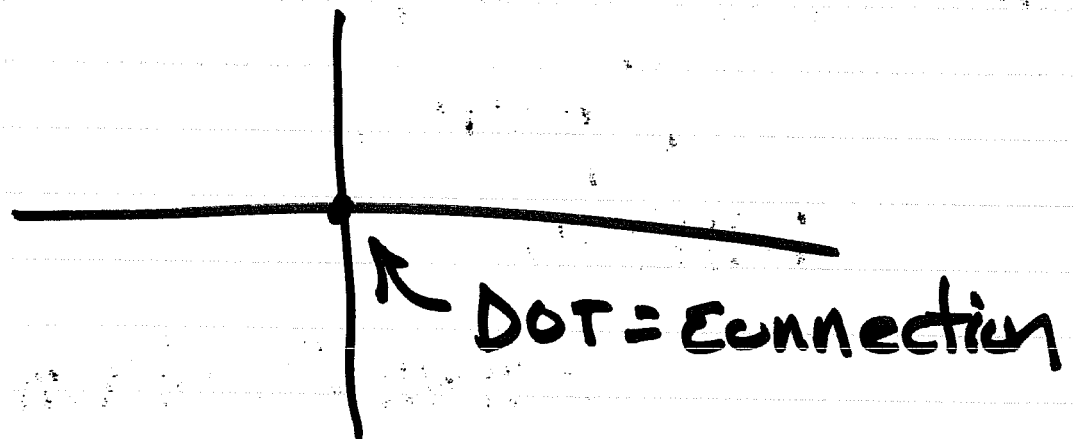
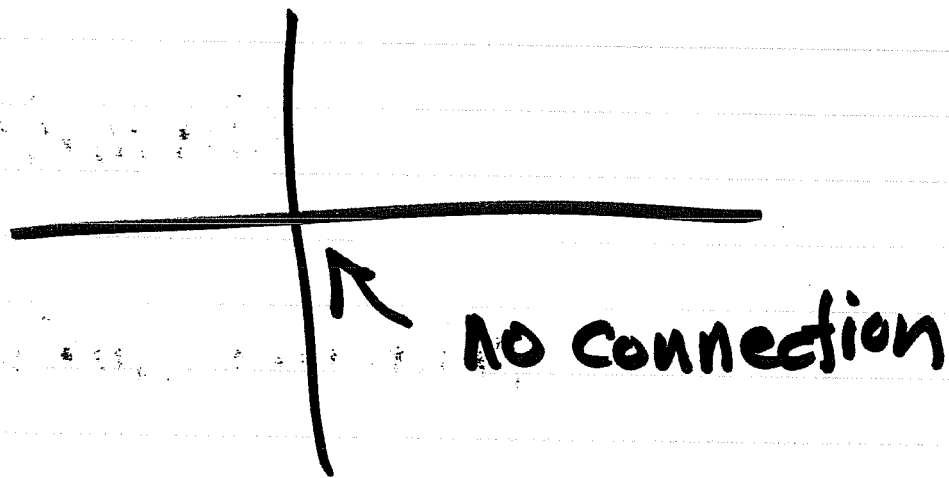
↳ VHDL

Schematic Elements



↗ collection of wires(4) = bus


busses
buses



grammar = interpretation
of letters or
symbols to represent
something

Backus-Naur Format - BNF

D_3 D_2 D_1 D_0

$D_{3:0}$ \swarrow MSB \nwarrow LSB 

$D_{0:3}$ 

Interpretation
faces

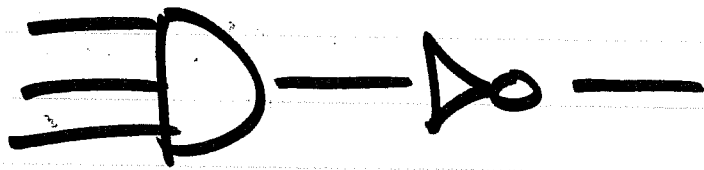
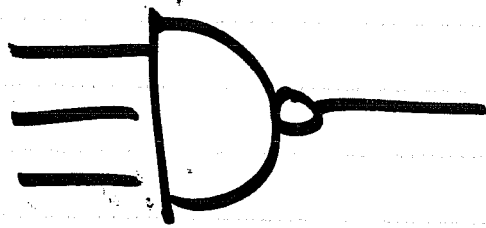
$D[3:0]$

$D[3..0]$

Boolean Algebra
(Claude Shannon)

De Morgan's Theorem

B · C · D



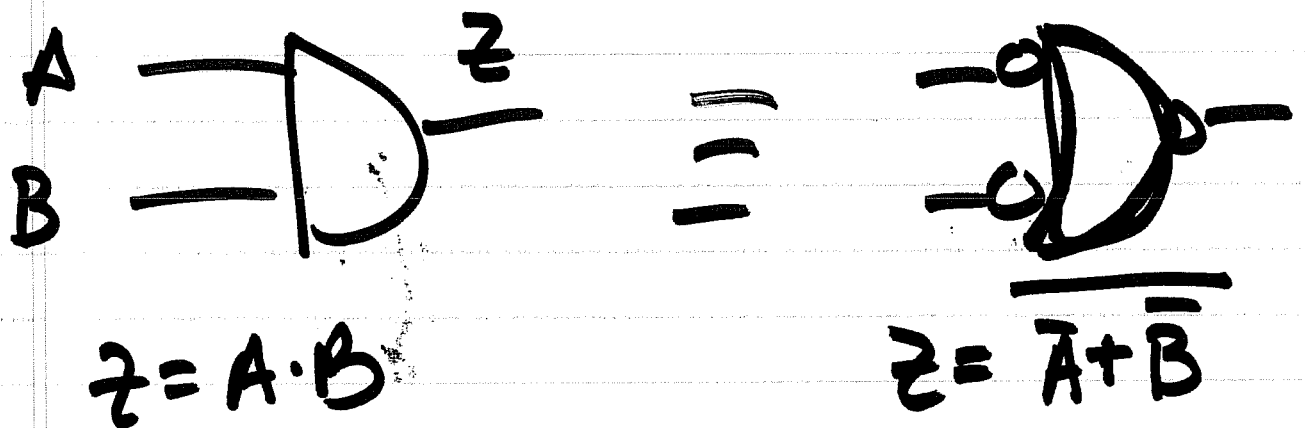
~~B C D~~

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

$$\overline{\overline{CD}} = CD$$

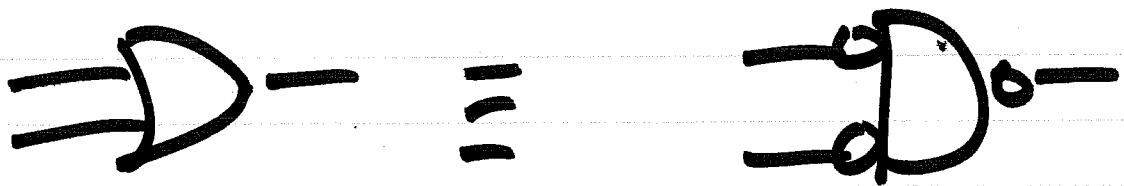
$$\overline{\cancel{CD}} = \overline{\bar{C} + \bar{D}}$$

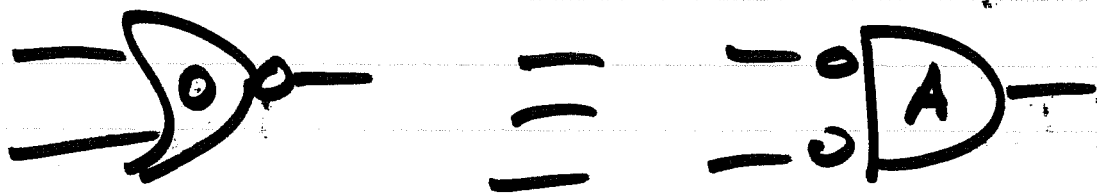
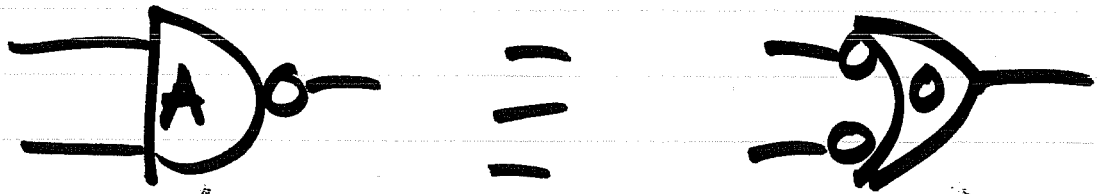
$$\overline{\bar{C} + \bar{D}} \rightarrow \cancel{\overline{\bar{C} + \bar{D}}} \\ \overline{\bar{C}} \cdot \overline{\bar{D}} \\ C \cdot D$$



$$= \overline{\overline{A} \cdot \overline{B}}$$

$$= A \cdot B$$



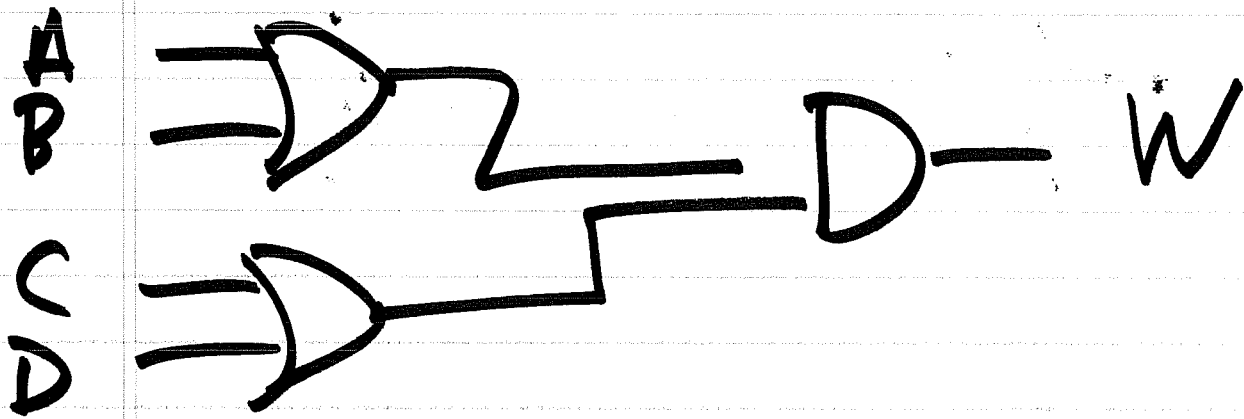


Be
careful

$$\overline{(A \cdot B) + C} = \overline{(A \cdot B)} \cdot \overline{C}$$

$$= (\overline{A} + \overline{B}) \cdot \overline{C}$$

order s/B
maintained



- 1.) start at output
- 2.) work back to DeMorgan backwards
- 3.) stop when you want to

Bubble Pushing

