COMP311: *COMPUTER ORGANIZATION!*

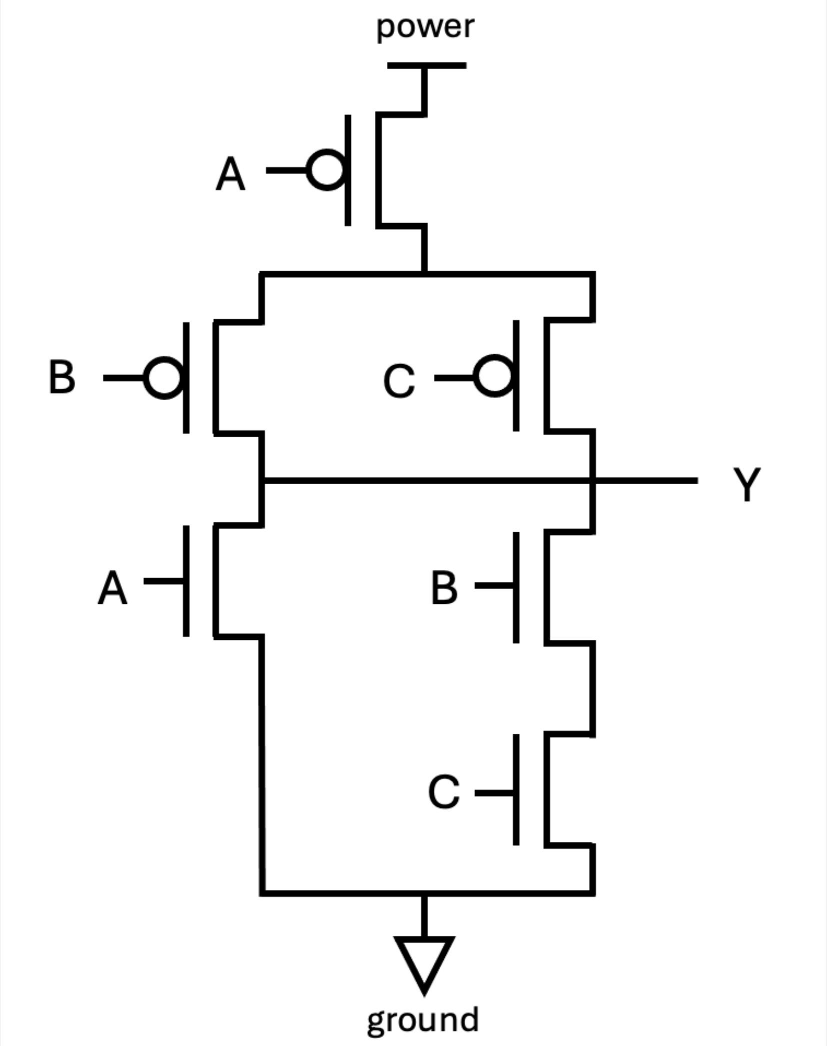
Lecture 5: Logic Gates

tinyurl.com/comp311-fa25

Logistics Update

- First Written Assignment due Thursday 11:59pm
  - *Come to office hours!*
  - *If you turn it in by Wednesday at 5pm, we can get you graded feedback before the quiz.*
- First Quiz on Thursday. Topics will include:
  - *211 review*
    - Going between decimal, binary, hex
    - Two's complement,
    - Overflow
  - *Transistors*
  - *Boolean Algebra Equations + Logic Diagrams* (TODAY!)
- Review session tomorrow at 7:30pm (location TBD)

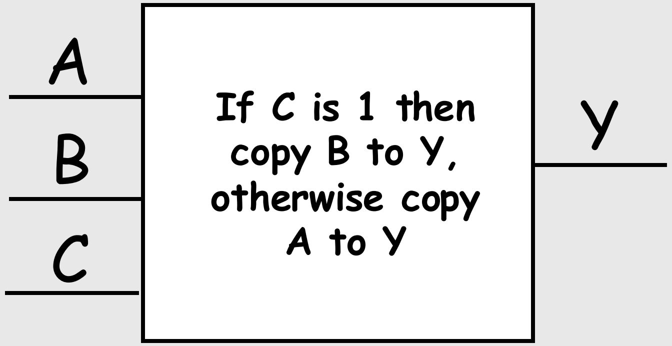
Refresher: Transistor Tracing



LOGIC GATES

Now can we design larger systems!

We need to start somewhere – usually with a functional specification



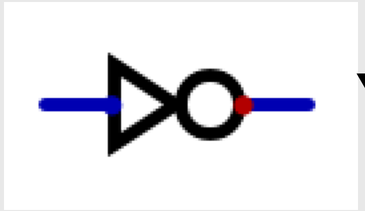
****Every combinational function can be expressed as a table**!**

"**Truth tables**" are a concise description of the combinational system's function, where an output is specified for *every* input combination.

Inverter/Not Gate

n inputs
⇒ rows TT
2^n

Symbol

A  Y

Truth Table

A	Y
0	1
1	0

Equation

$$Y = \overline{A}$$

$2^1 \Rightarrow$ 2 rows ✓

What Gates can we build?

Recall, we need to design our gates using a pull-up network of pMOS transistors and a pull-down network of nMOS transistors

What gates can we
- build?
- define?

Let's start by considering only 2-input gates.

AND

AB	Y
00	0
01	0
10	0
11	1

OR

AB	Y
00	0
01	1
10	1
11	1

NAND

AB	Y
00	1
01	1
10	1
11	0

NOR

AB	Y
00	1
01	0
10	0
11	0

How many possible 2-input gates are there?

KEY IDEA: As many as there are 2-input truth tables.

All the gates!

There are only 16 possible 2-input gates… Let's examine all of them. Some we already know, others are just silly.

INPUT AB	ZERO	AND	A > B	A	B > A	B	XOR	OR	NOR	XNOR	NOT 'B'	A <= B	NOT 'A'	B <= A	NAND	ONE
00	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
01	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
10	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
11	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

How many of these gates can be implemented using a single CMOS gate?

Do we really need all of these gates?

Nope! Once we realize that we can describe all of them using just AND, OR, and NOT

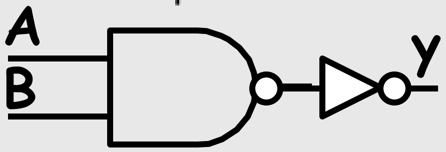
Composing gates: AND and OR

Each can be constructed using a pair of CMOS gates

AND is just NAND with an inverter, and OR is just NOR with an inverted output.

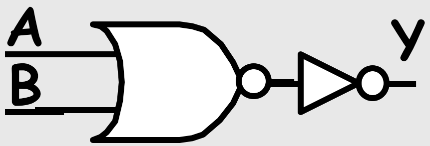
AND

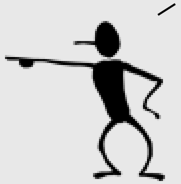
AB	Y
00	0
01	0
10	0
11	1



OR

AB	Y
00	0
01	1
10	1
11	1

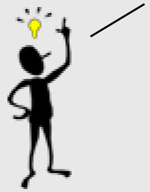




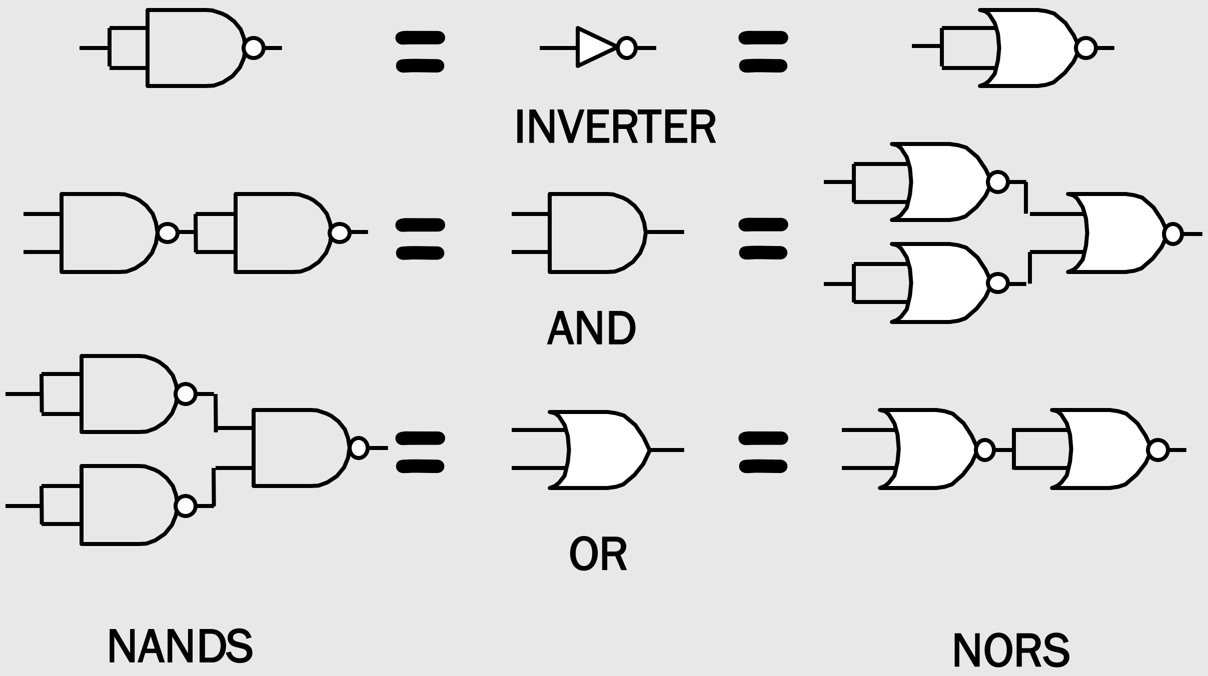
These two gates are particularly important. Using them will allows us to develop a systematic approach for constructing any combinational function.

One will do!

A UNIVERSAL gate is one that can be used to implement *ANY* COMBINATIONAL FUNCTION. There are many UNIVERSAL gates, but not all gates are UNIVERSAL.

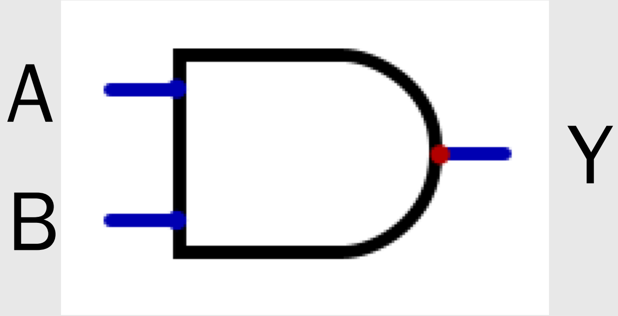


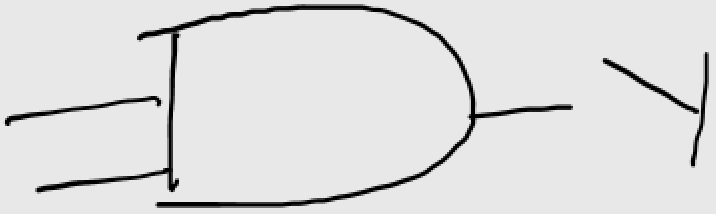
NANDs and NORs are UNIVERSAL!



AND Gate

Symbol





Truth Table

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

Equation

$$Y = A \times B$$

or

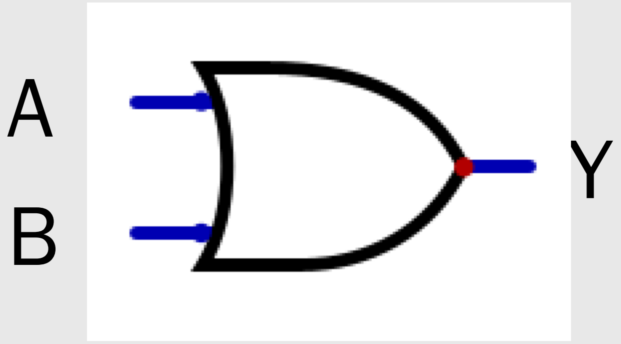
$$Y = AB$$

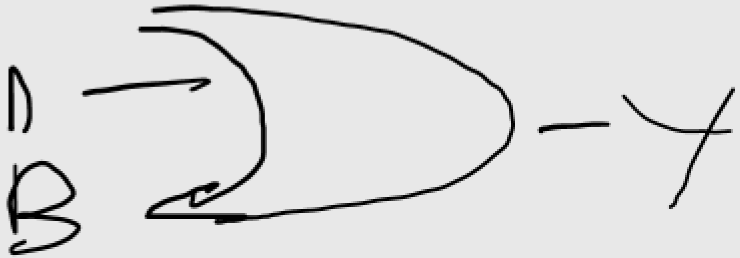
$$Y = A * B$$

$$Y = A \cdot B$$

OR Gate

Symbol





Truth Table

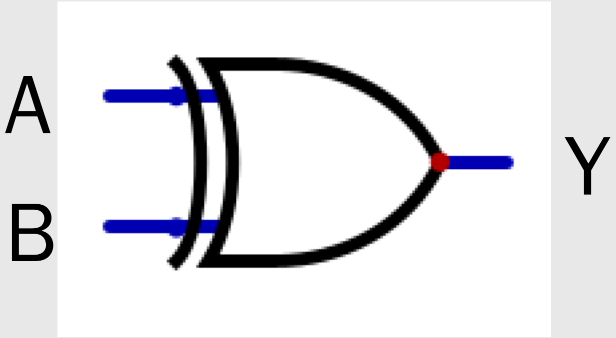
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

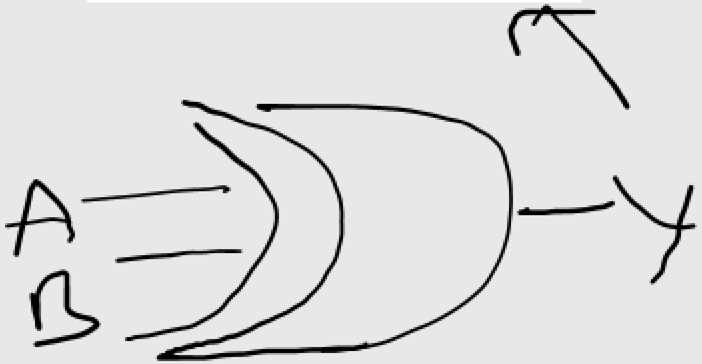
Equation

$$Y = A + B$$

XOR Gate

Symbol



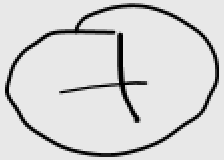


Truth Table

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

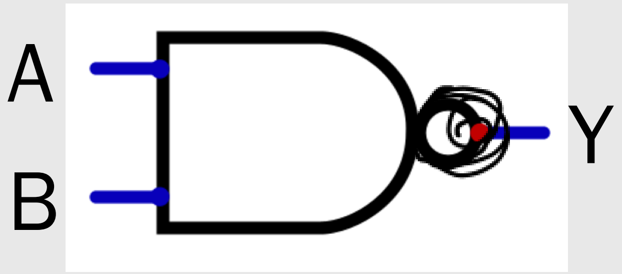
Equation

$$Y = A \oplus B$$



NAND

Symbol



Truth Table

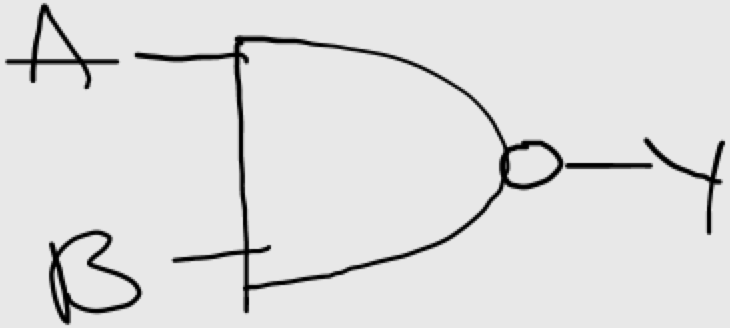
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

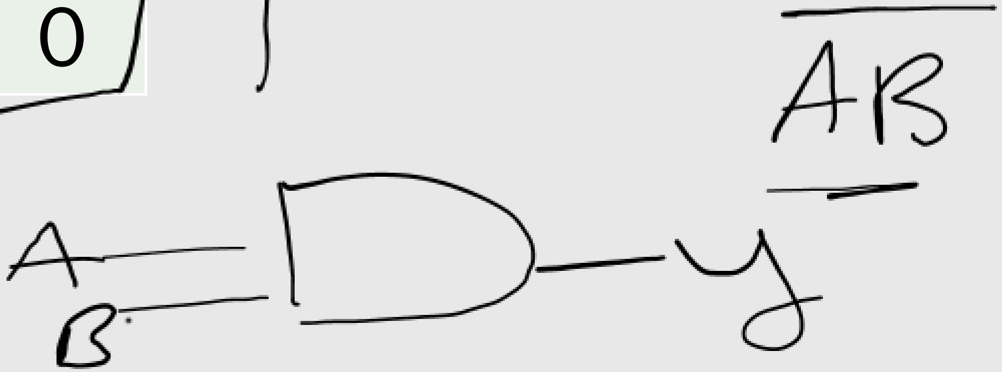
Equation

$$Y = \overline{A\,x\,B}$$

or

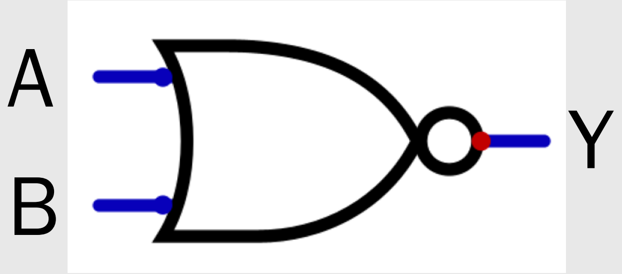
$$Y = \overline{AB}$$

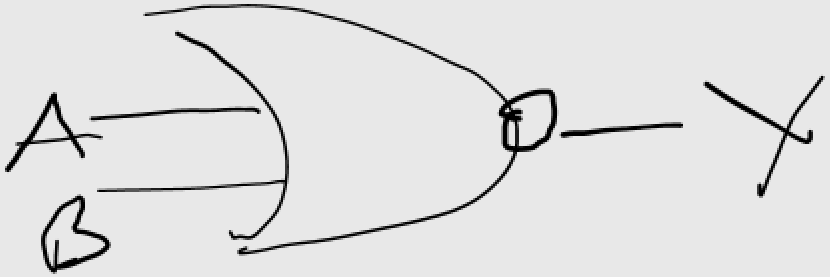




NOR

Symbol





Truth Table

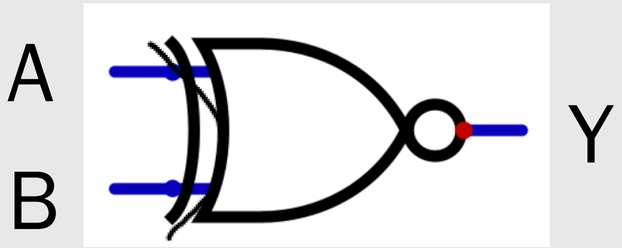
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

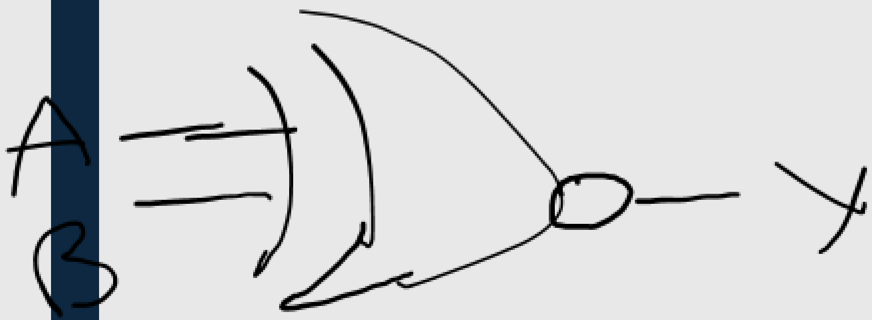
Equation

$$Y = \overline{A + B}$$

XNOR Gate

Symbol





Truth Table

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

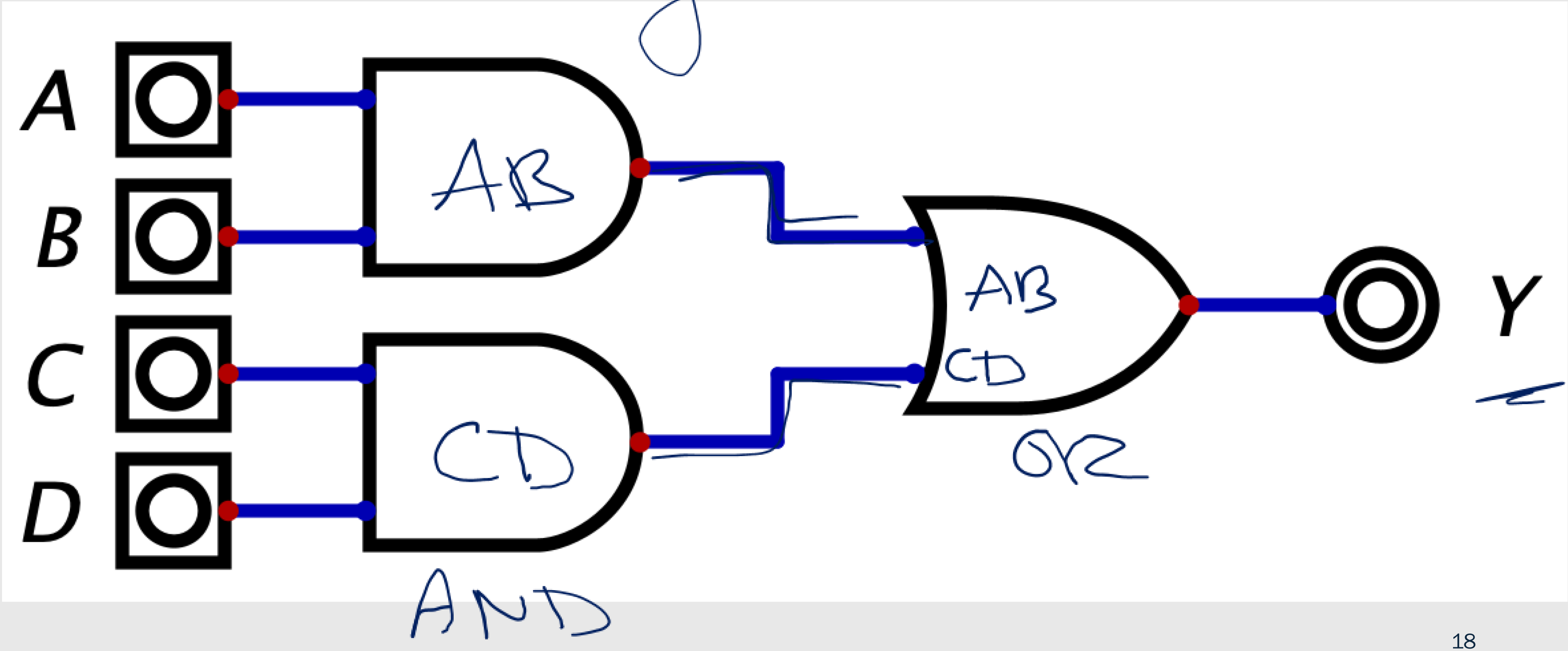
Equation

$$Y = \overline{A \oplus B}$$

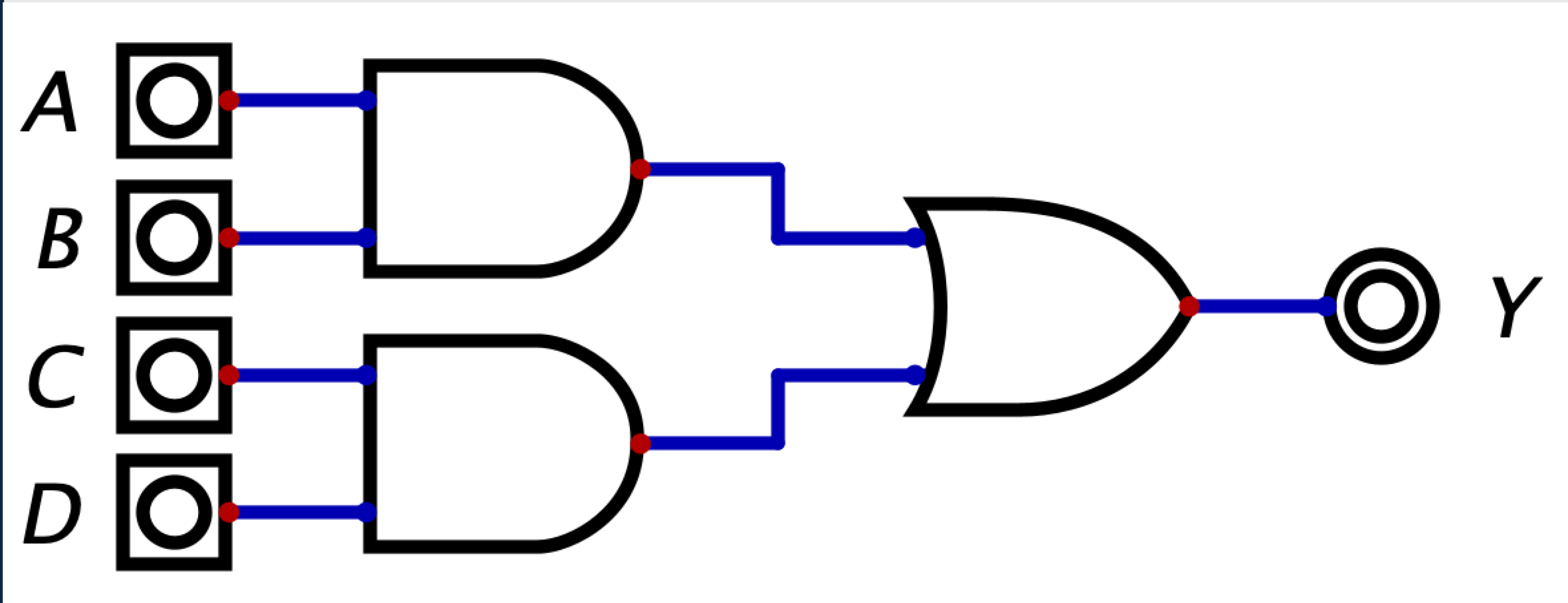
FORMING EQUATIONS FROM LOGIC DIAGRAMS

Ex1: Forming Equations from Logic Diagrams

y = AB + CD

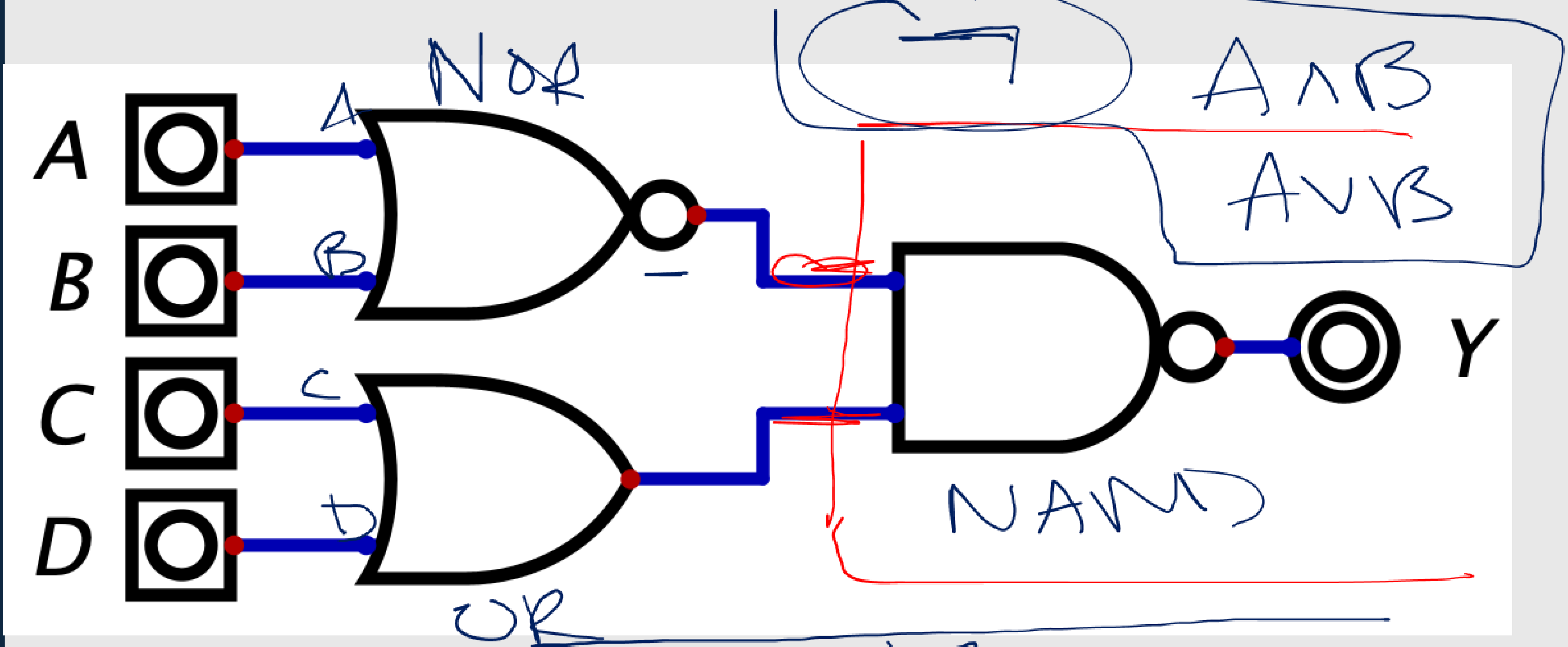


Ex1: Forming Equations from Logic Diagrams



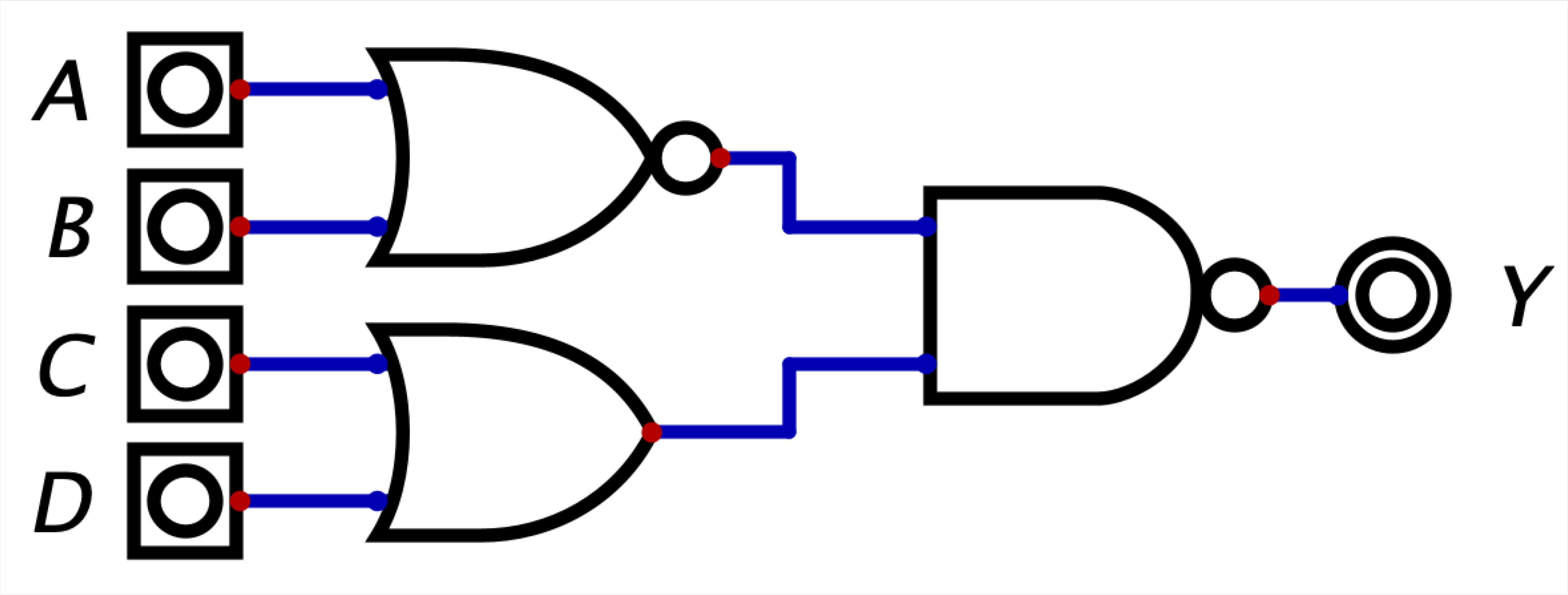
$$Y = AB + CD$$

Ex2: Forming Equations from Logic Diagrams



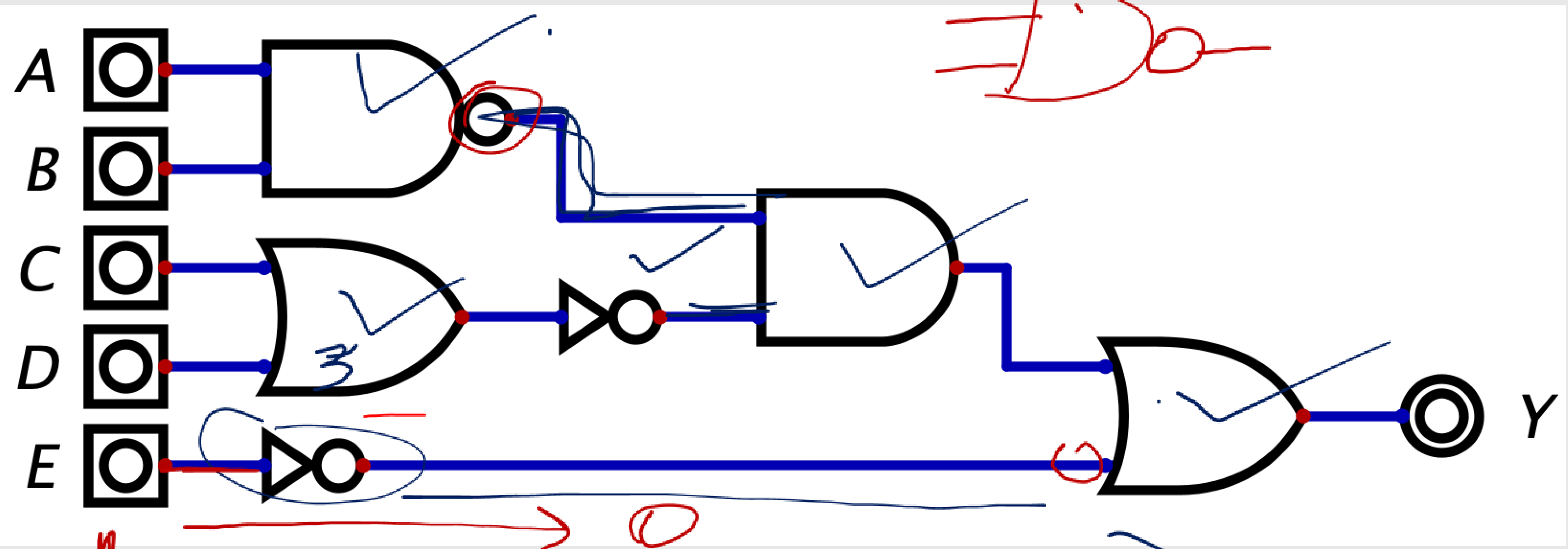
$\rightarrow y = \overline{(\overline{A+B})(C+D)}$

Ex2: Forming Equations from Logic Diagrams



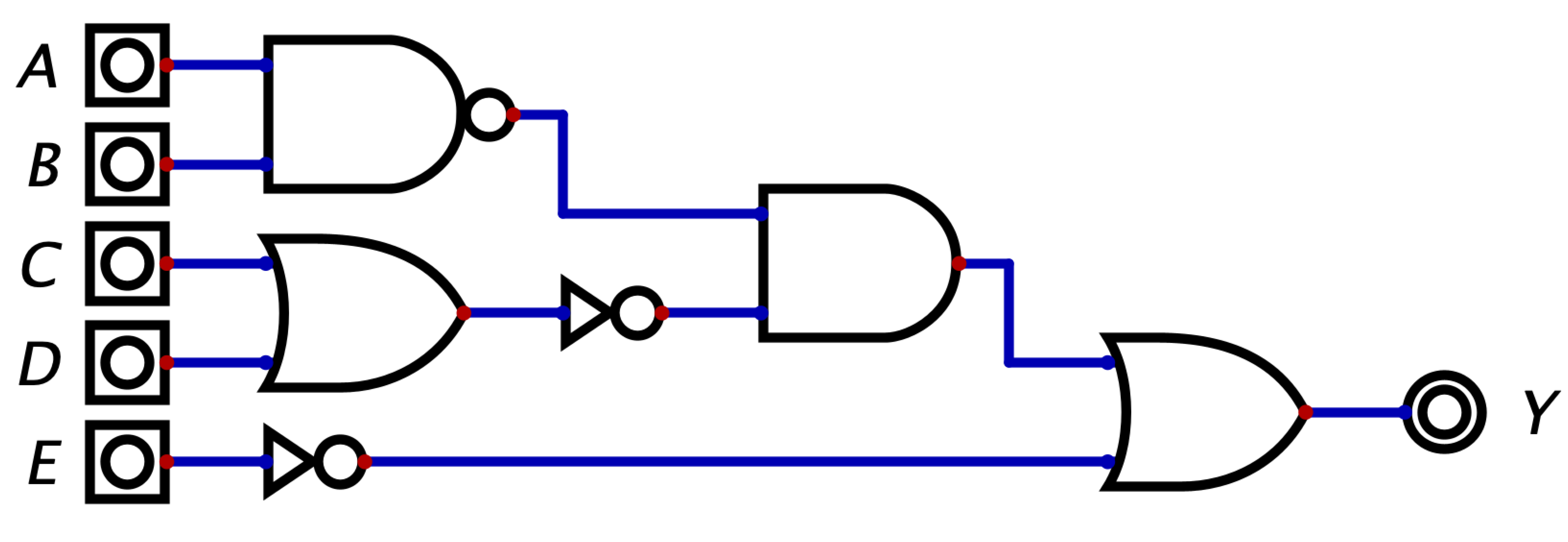
$$Y = \overline{(\overline{A + B})(C + D)}$$

Ex3: Forming Equations from Logic Diagrams



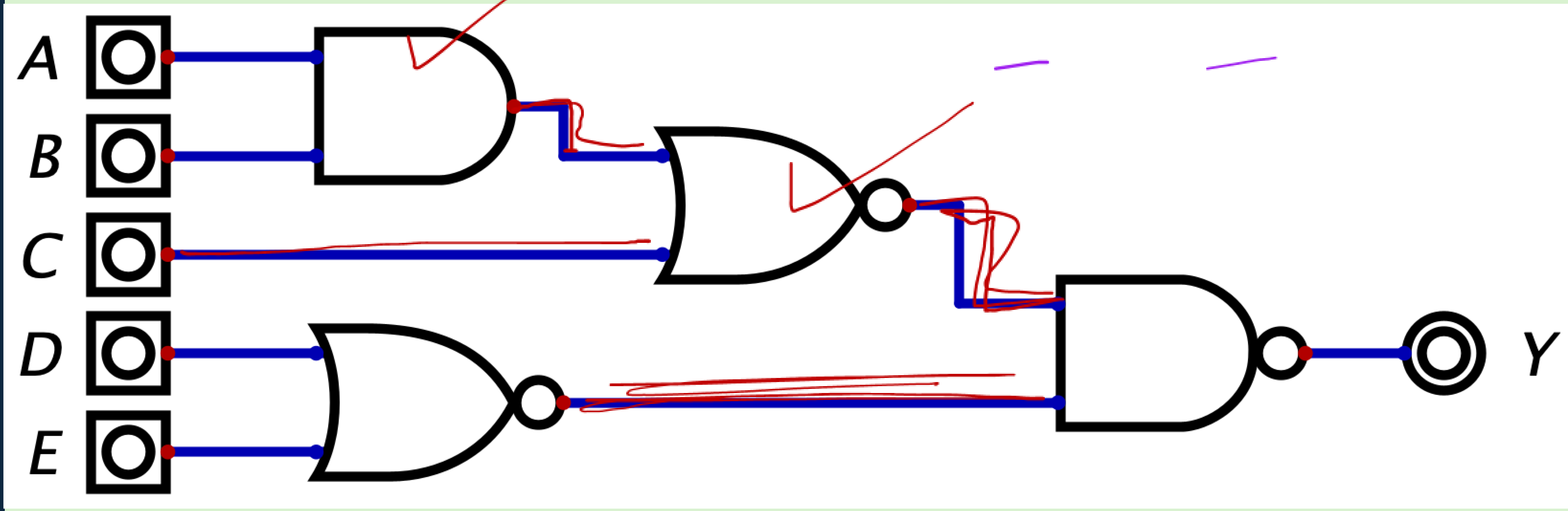
$Y = (\overline{AB})(\overline{C+D}) + \overline{E}$

Ex3: Forming Equations from Logic Diagrams



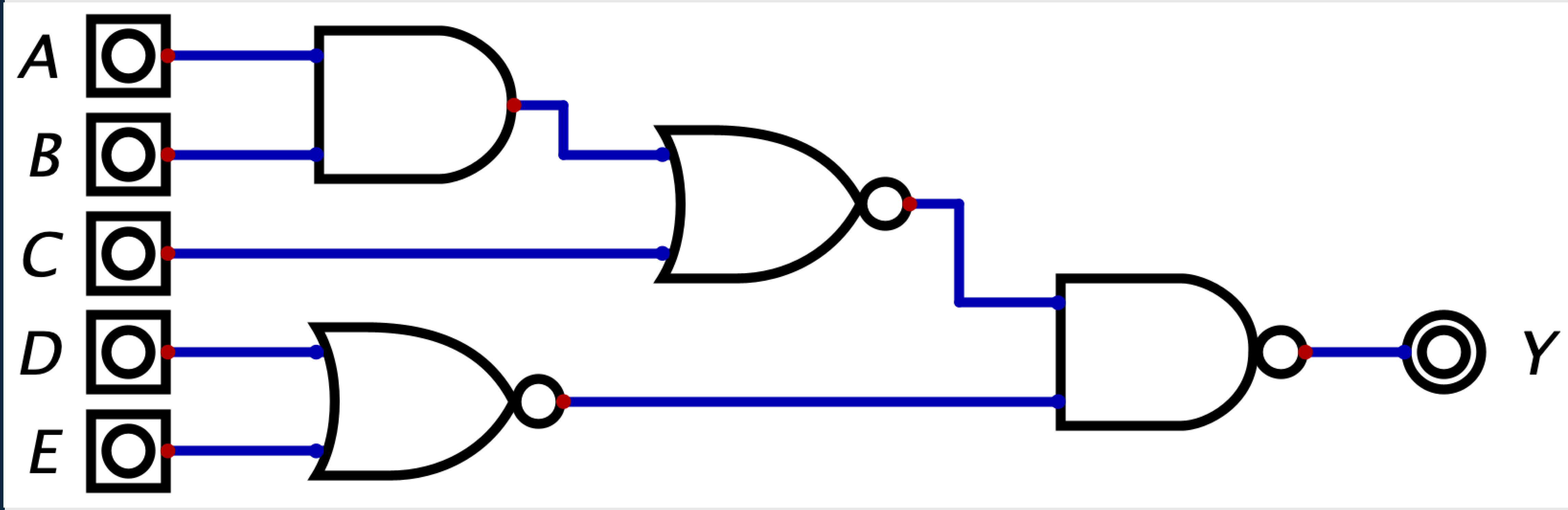
$$Y = (\overline{AB})\overline{(C + D)} + \bar{E}$$

EX4: Forming Equations from Logic Diagrams



$$y = \overline{\left(\overline{(AB)+C}\right)\left(\overline{D+E}\right)}$$

Forming Equations from Logic Diagrams

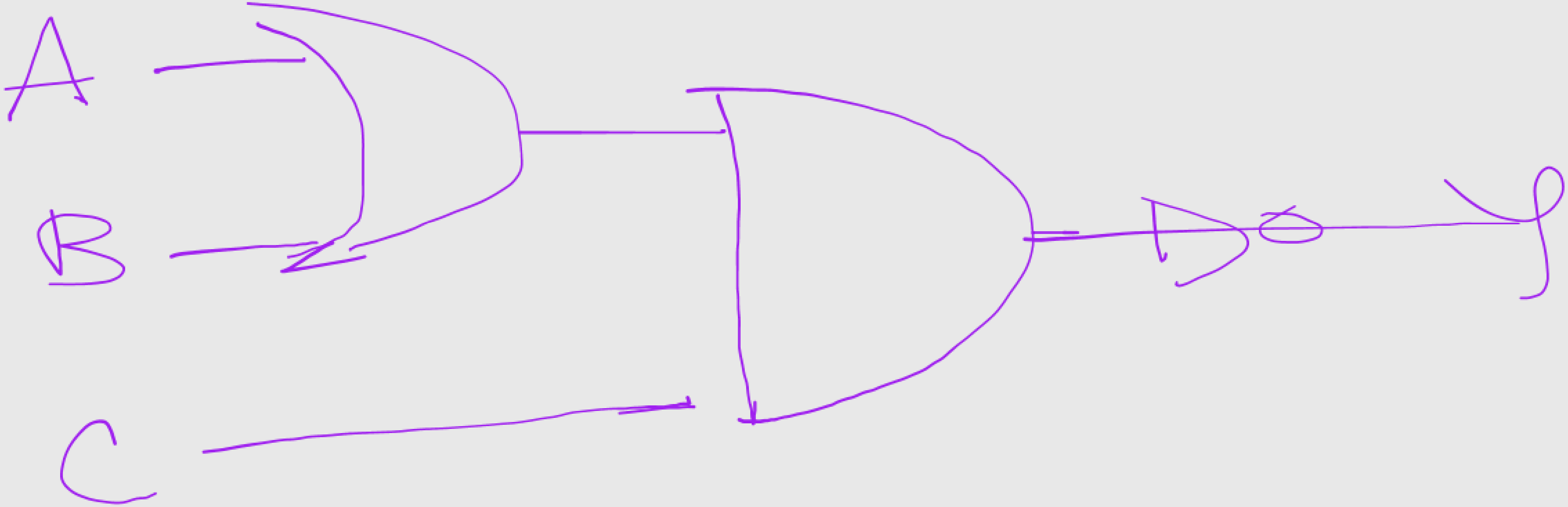


$$Y = \overline{\overline{(AB + C)}\ \overline{(D + E)}}$$

FORMING LOGIC DIAGRAMS FROM EQUATIONS

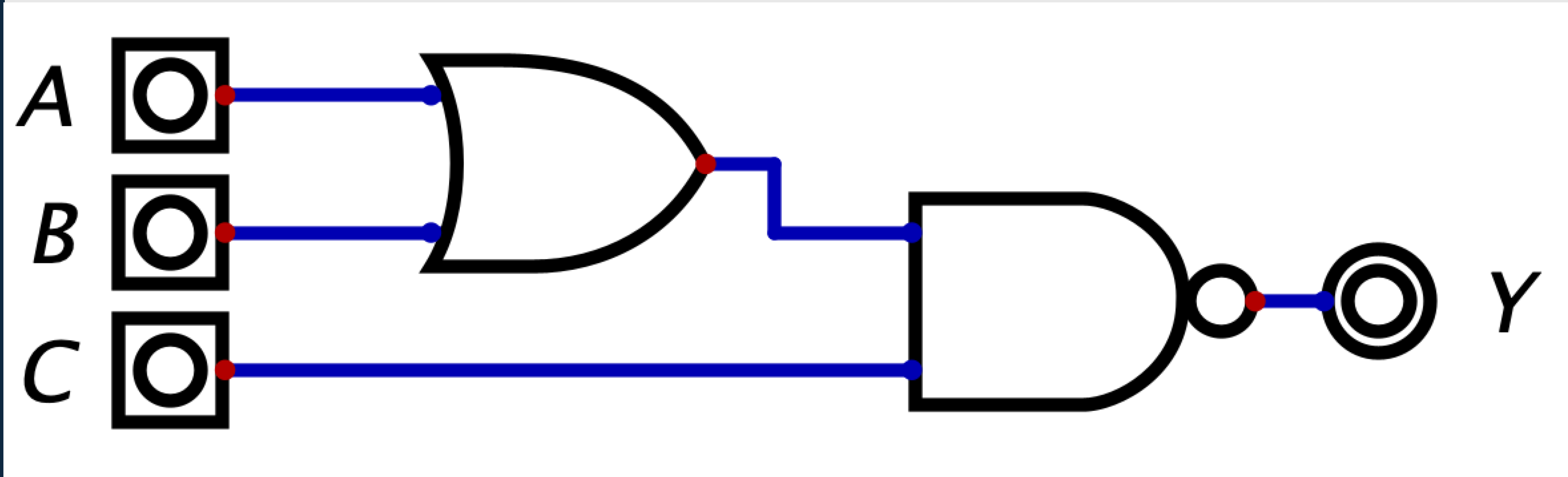
Ex1: Forming Logic Diagrams from Equations

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



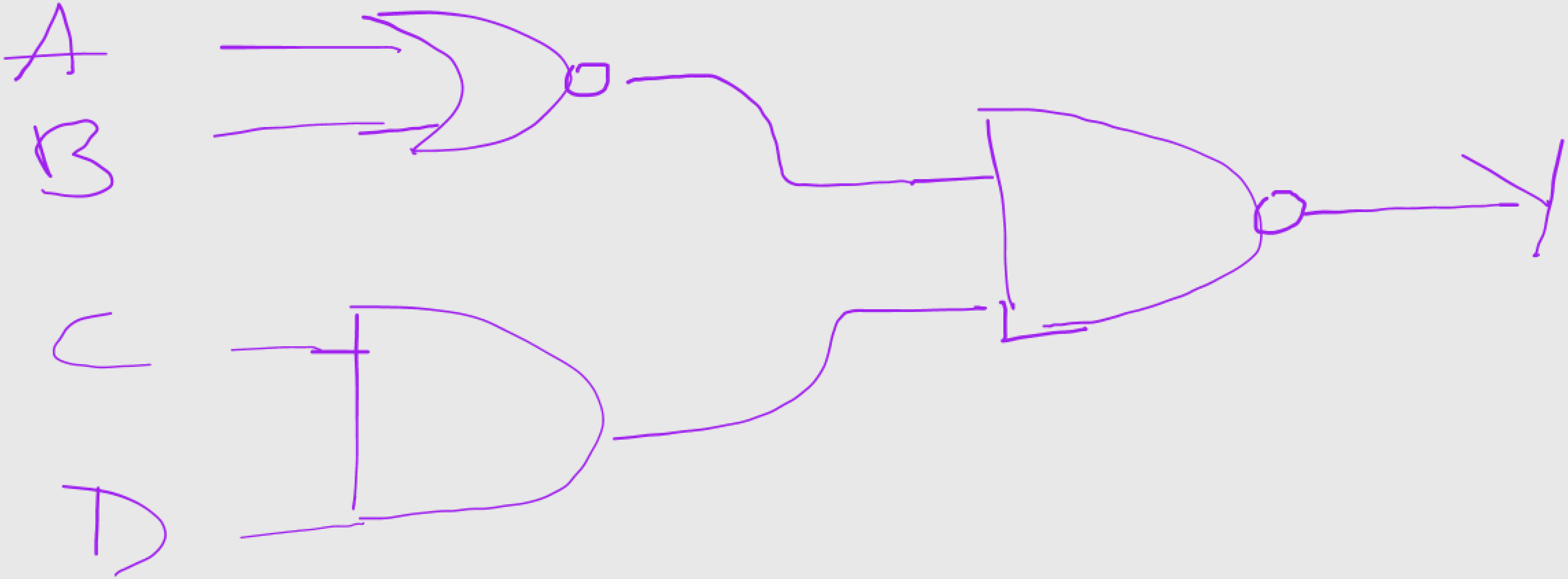
Ex1: Forming Logic Diagrams from Equations

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

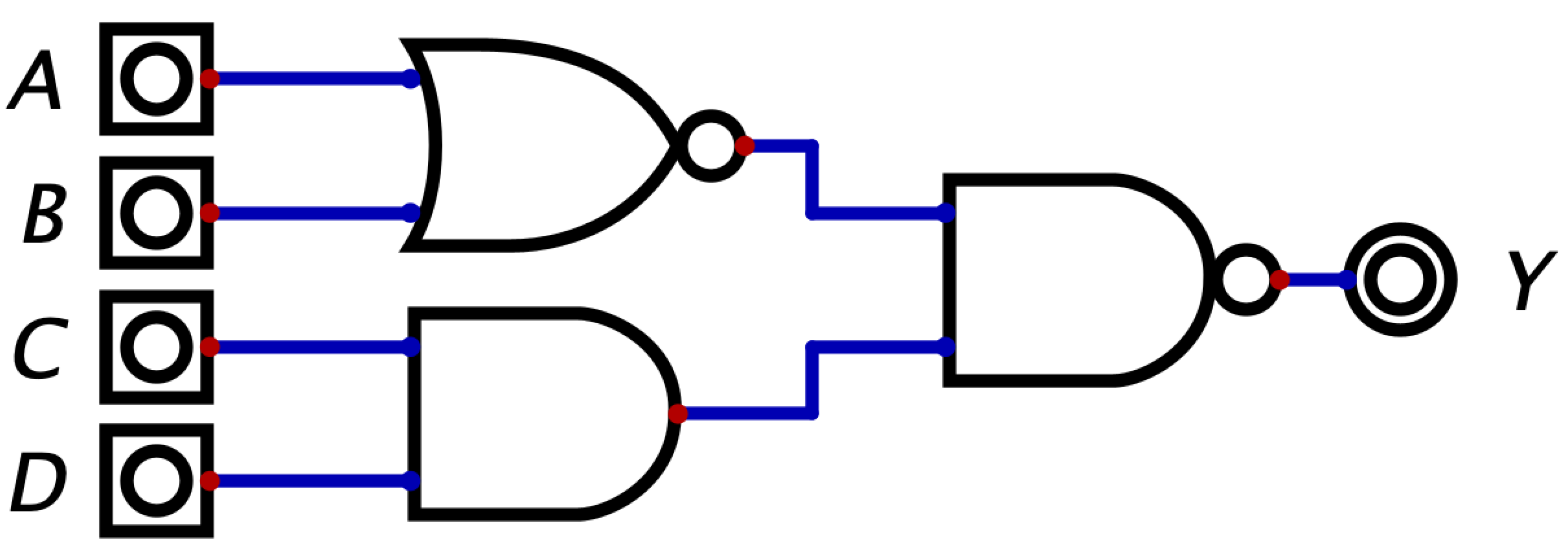


Ex2: Forming Logic Diagrams from Equations

$$Y = \overline{\overline{(A+B)}CD}$$

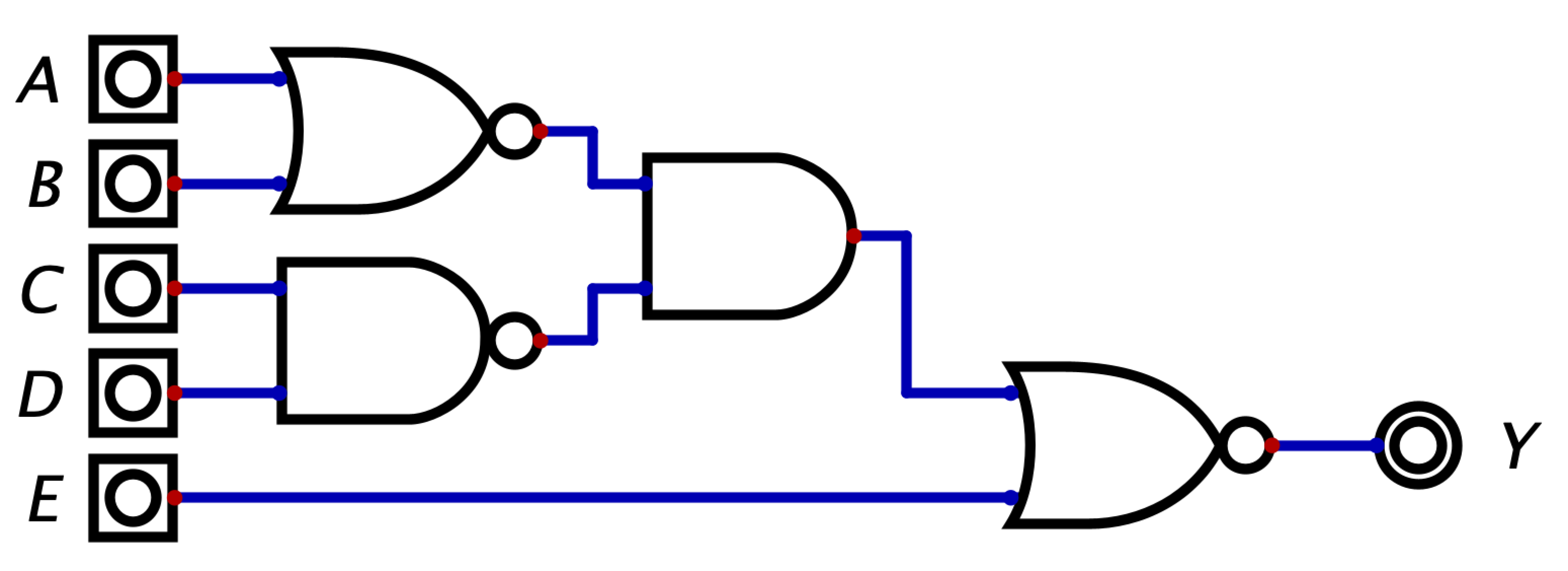


Ex2: Forming Logic Diagrams from Equations $Y = \overline{\overline{(A + B)}CD}$



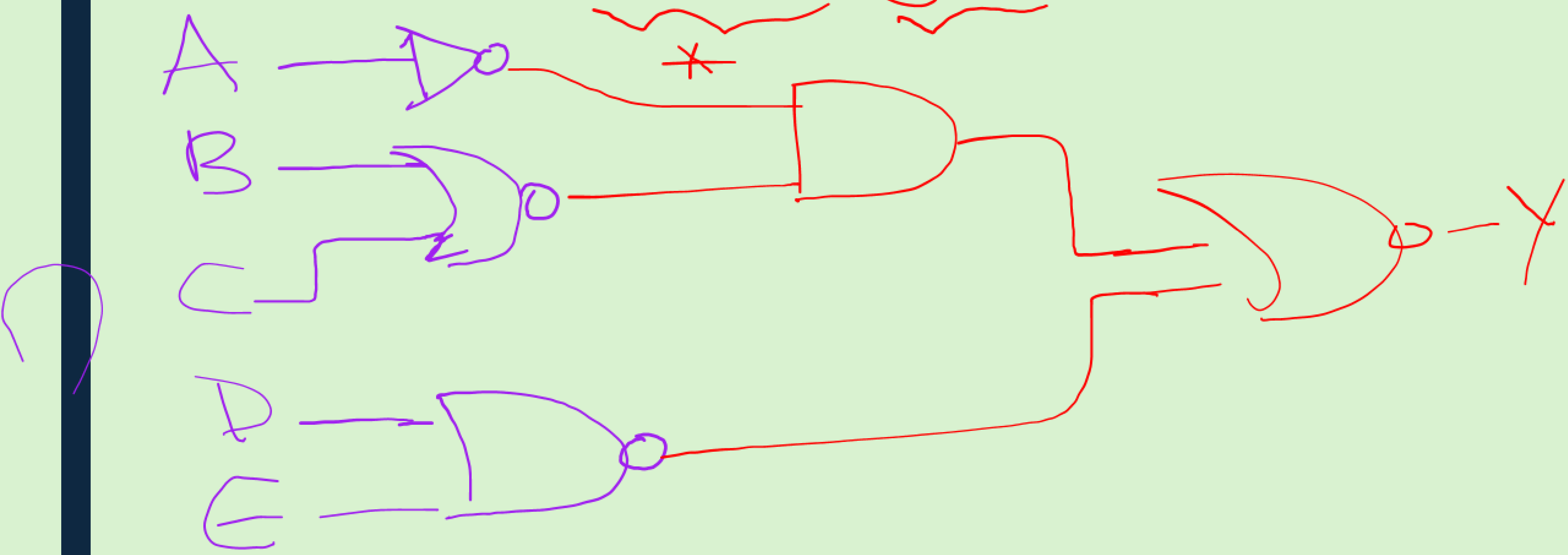
Ex3: Forming Logic Diagrams from Equations $Y = \overline{(\overline{A + B})(\overline{CD}) + E}$

Ex3: Forming Logic Diagrams from Equations $Y = \overline{(\overline{A+B})(\overline{CD}) + E}$



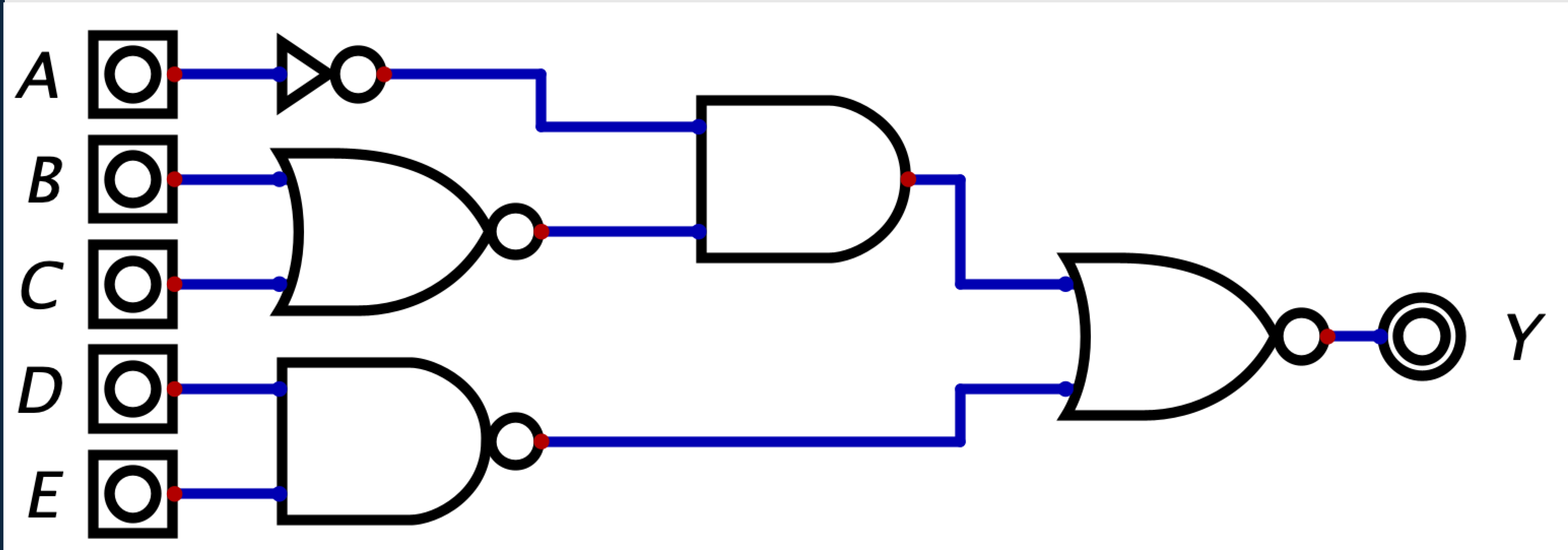
EX4: Forming Logic Diagrams from Equations

$$Y = \overline{(\bar{A})(\overline{B + C}) + \overline{DE}}$$



Forming Logic Diagrams from Equations

$$Y = \overline{(\bar{A})(\overline{B + C}) + \overline{DE}}$$



FORMING TRUTH TABLES FROM EQUATIONS

Ex1: Forming Truth Tables from Equations

$$Y = A + \bar{B}$$

→ 1 + 0 ⇒

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

2 inputs ⇒ 4 rows

Ex1: Forming Truth Tables from Equations

$$Y = A + \bar{B}$$

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

3 inputs ⟶ 8 rows

Ex2: Forming Truth Tables from Equations

$$Y = AB + \bar{C}$$

A	B	C	Y
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	1

Ex2: Forming Truth Tables from Equations

$$Y = AB + \bar{C}$$

A	B	C	Y
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	1

Ex3: Forming Truth Tables from Equations

$$Y = A + \bar{A}B\bar{C}$$

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

Ex3:Forming Truth Tables from Equations

$$Y = A + \bar{A}B\bar{C}$$

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

Ex4: Forming Truth Tables from Equations

$$Y = \bar{A}\bar{B} + \overline{AB}$$

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

Ex4: Forming Truth Tables from Equations

$$Y = \bar{A}\bar{B} + \overline{AB}$$

A	B	C	Y
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	0
1	1	1	0