UNIVERSALITY

UNIVERSAL SET

Is it possible to implement ALL the possible Boolean functions using NOT, AND, OR, NAND, NOR? Yes!

UNIVERSAL SET

What if we are not given some!
What if some are very costly! E.g., NOT
Can we reduce this set? E.g., building NOT by NAND/ NOR

SET	UNIVERSAL SET
{NOT, AND, OR, NAND, NOR}	Yes! (a Full Set)
{NOT, AND, OR}	Yes, (a Full Set) NAND = NOT-AND NOR = NOT-OR
{NOT, AND}	If we could design OR
{NOT, OR}	If we could design AND
{NOT}	If we could design AND, OR
{AND}	If we could design NOT, OR
{OR}	If we could design NOT, AND
{NAND}	If we could design NOT, AND, OR
{NOR}	If we could design NOT, AND, OR

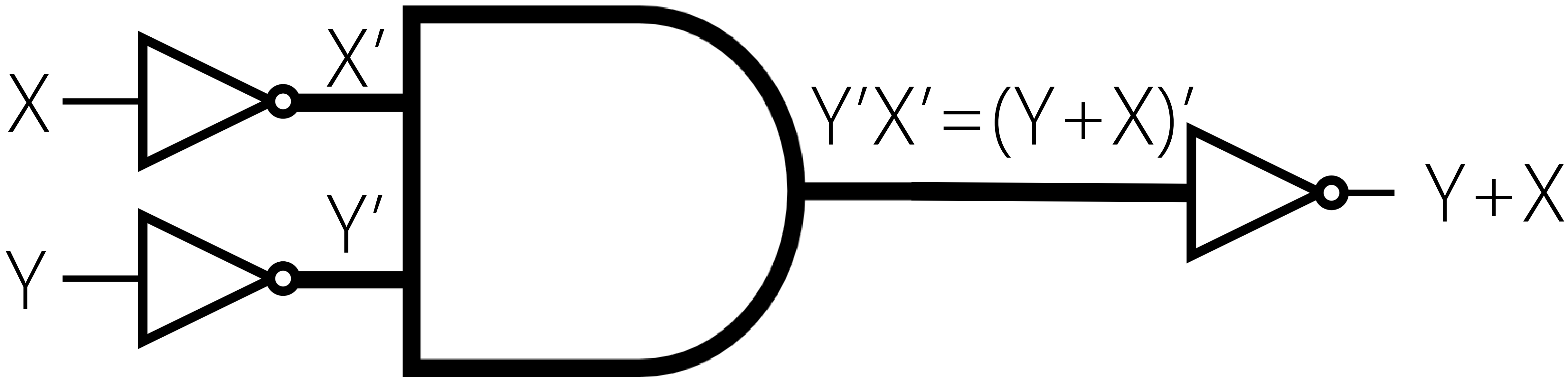
SET	UNIVERSAL SET
{NOT, AND, OR, NAND, NOR}	Yes! (a Full Set)
{NOT, AND, OR}	Yes, (a Full Set) NAND = AND-NOT NOR = OR-NOT
{NOT, AND}	If we could design OR (a Complete Set)
{NOT, OR}	If we could design AND (a Complete Set)
{NOT}	If we could design AND, OR
{AND}	If we could design NOT, OR
{OR}	If we could design NOT, AND
{NAND}	If we could design NOT, AND, OR
{NOR}	If we could design NOT, AND, OR

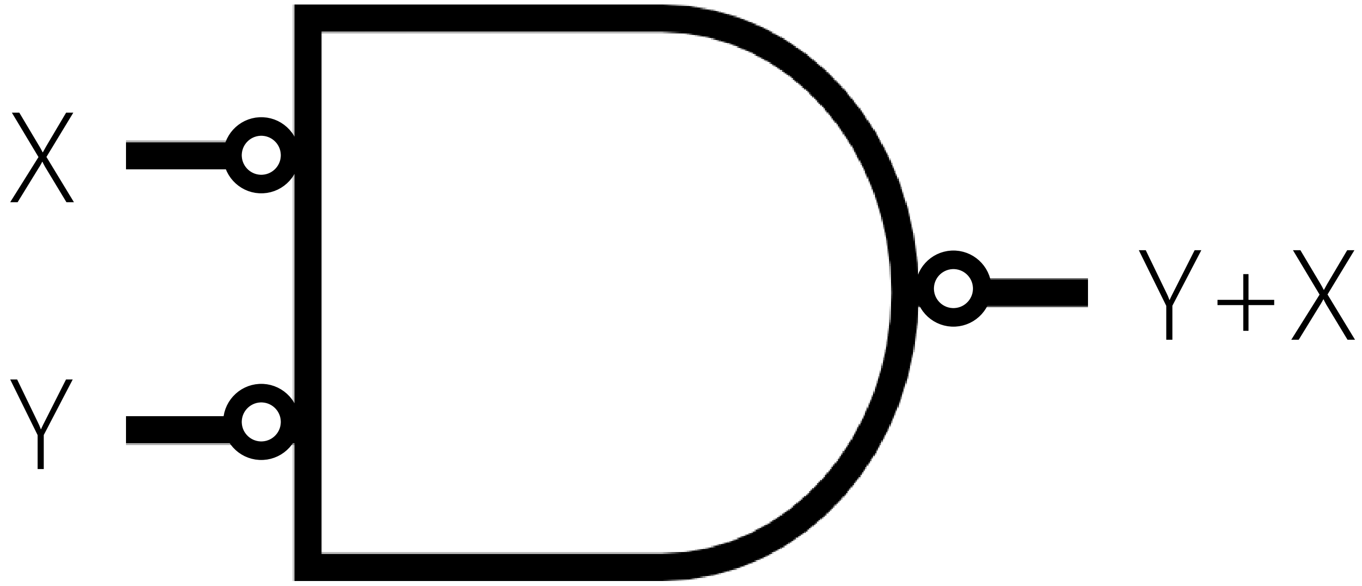
UNIVERSAL SET
{NOT, AND}

Augustus De Morgan (1806–1871) Mathematician Logician

DE MORGAN'S LAWS

- $(Y+X)' = Y'X'$


X
Y
X′
Y′
Y′X′=(Y+X)′
Y+X



OR

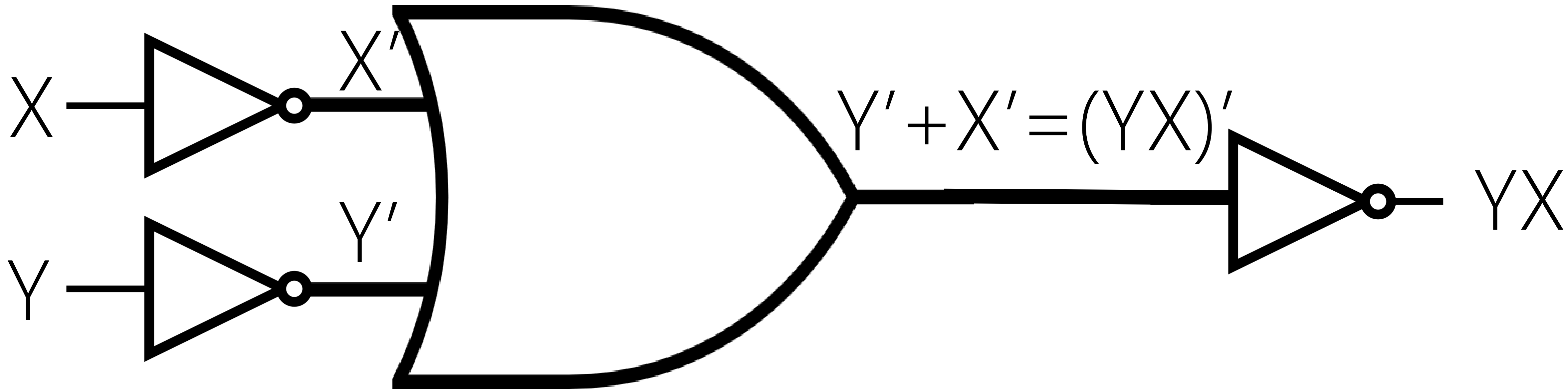
SET	UNIVERSAL SET
{NOT, AND, OR, NAND, NOR}	Yes! (a Full Set)
{NOT, AND, OR}	Yes, (a Full Set) NAND = AND-NOT NOR = OR-NOT
{NOT, AND}	If we could design OR (a Complete Set)
{NOT, OR}	If we could design AND (a Complete Set)
{NOT}	If we could design AND, OR
{AND}	If we could design NOT, OR
{OR}	If we could design NOT, AND
{NAND}	If we could design NOT, AND, OR
{NOR}	If we could design NOT, AND, OR

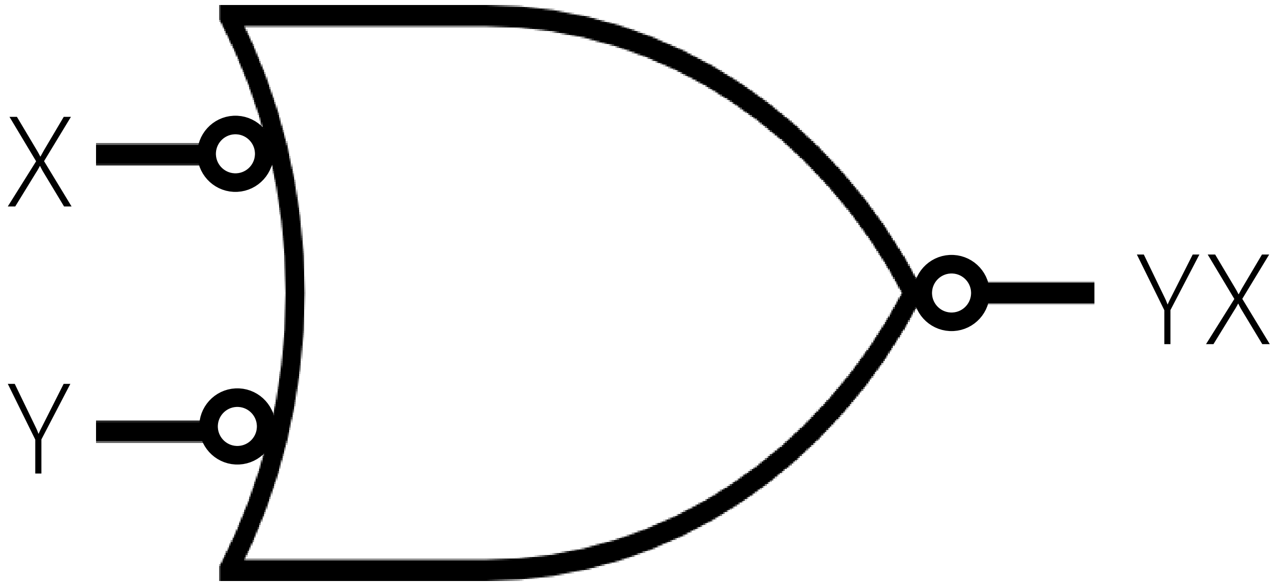
UNIVERSAL SET
{NOT, OR}

Augustus De Morgan
(1806–1871)
Mathematician
Logician

DE MORGAN'S LAWS

- $Y' + X' = (YX)'$


X
Y
X′
Y′
Y′+X′=(YX)′
YX

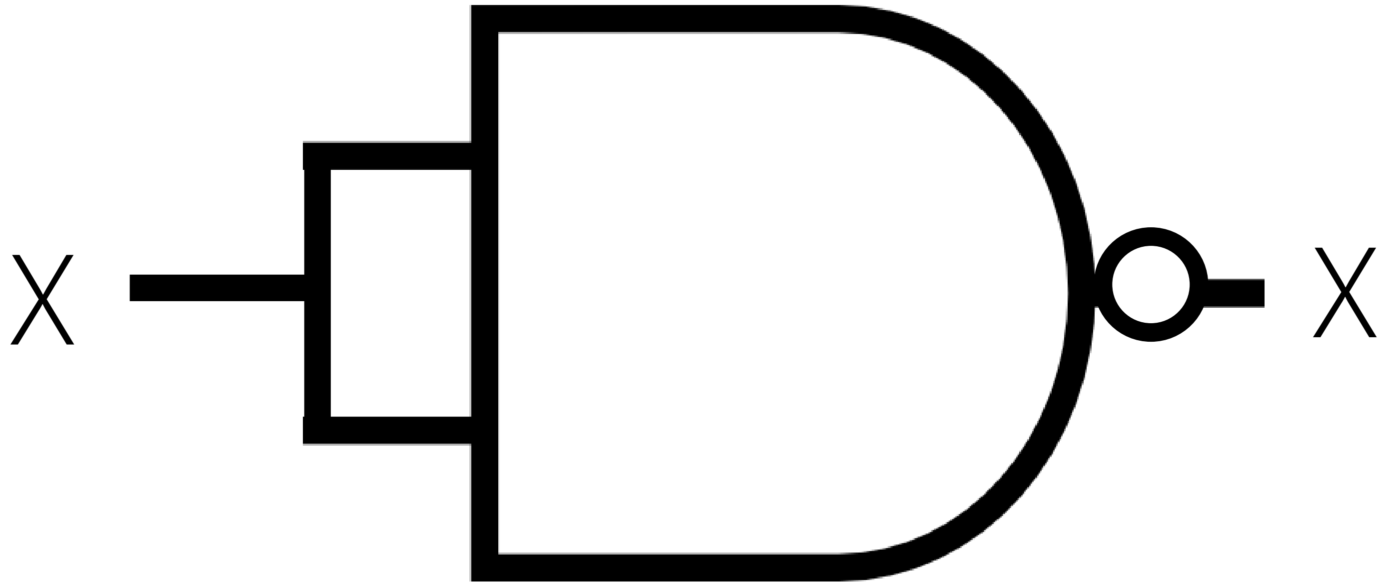


AND

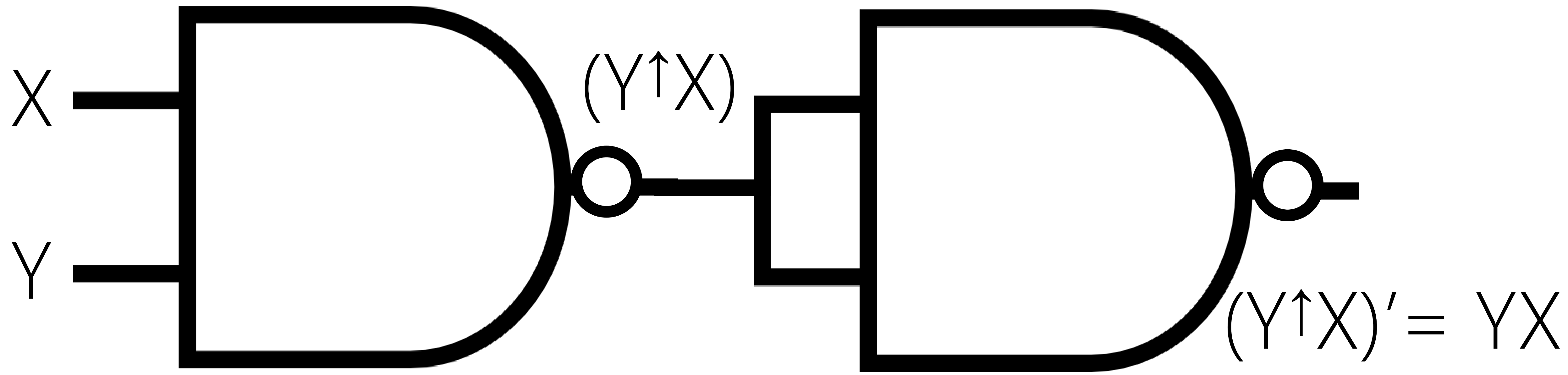
SET	UNIVERSAL SET
{NOT, AND, OR, NAND, NOR}	Yes! (a Full Set)
{NOT, AND, OR}	Yes, (a Full Set) NAND = AND-NOT NOR = OR-NOT
{NOT, AND}	If we could design OR (a Complete Set)
{NOT, OR}	If we could design AND (a Complete Set)
{NOT}	If we could design AND, OR
{AND}	If we could design NOT, OR
{OR}	If we could design NOT, AND
{NAND}	If we could design NOT, AND, OR
{NOR}	If we could design NOT, AND, OR

UNIVERSAL GATE
{NAND}

NOT ► $(XX)' = (X \uparrow X) = X'$



AND►NOT (NAND) = $((Y^{\uparrow}X))' = YX$



OR ► DE MORGAN'S LAW

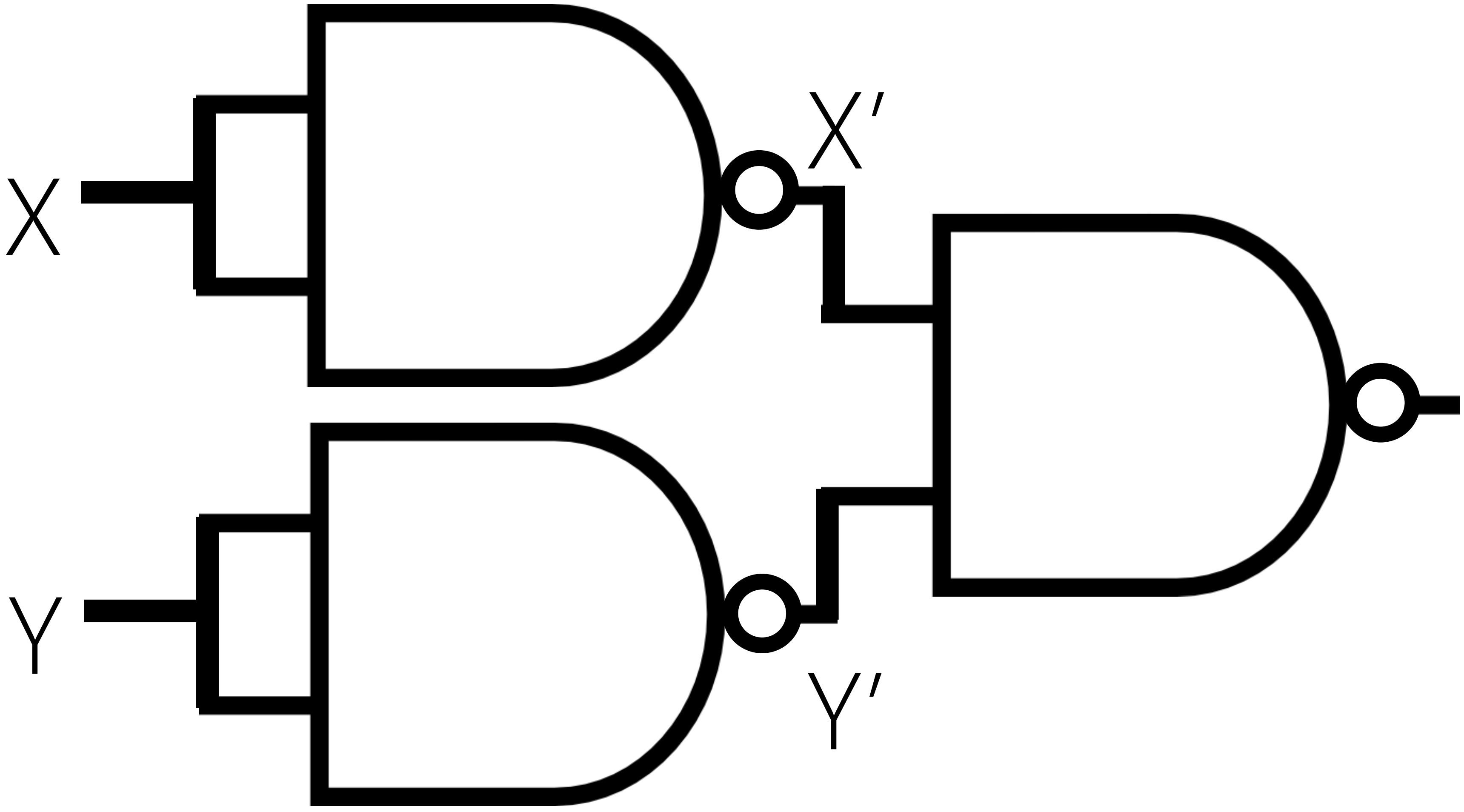
$$(Y+X)' = Y'X'$$

$$((Y+X)')' = (Y'X')'$$

$$Y + X = (Y'X')'$$

$$Y + X = Y' \uparrow X'$$

OR: DE MORGAN'S LAW



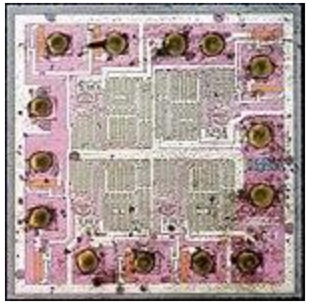
(Y′X′)′ =

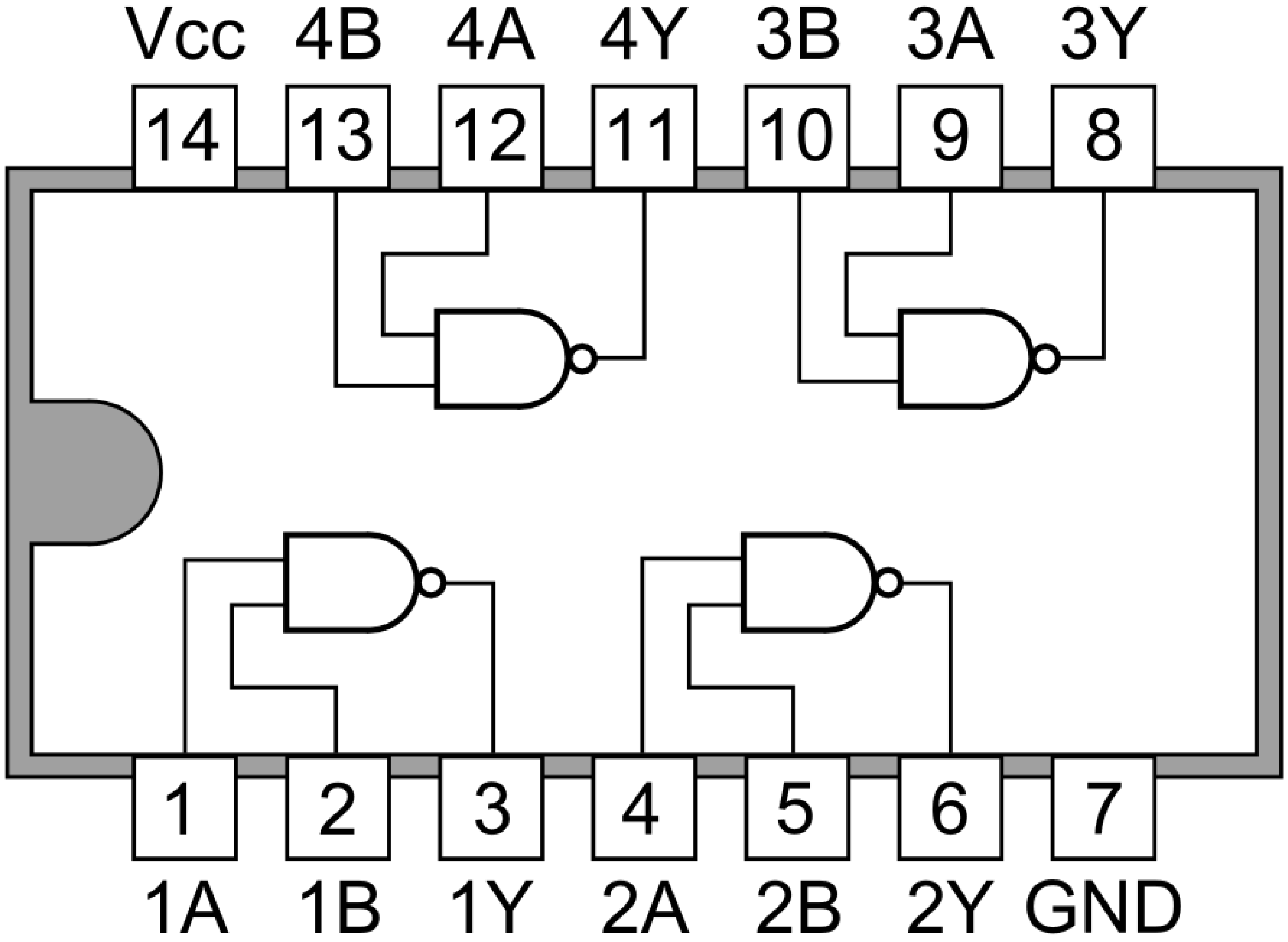
Y+X =

Y′ ↑ X′

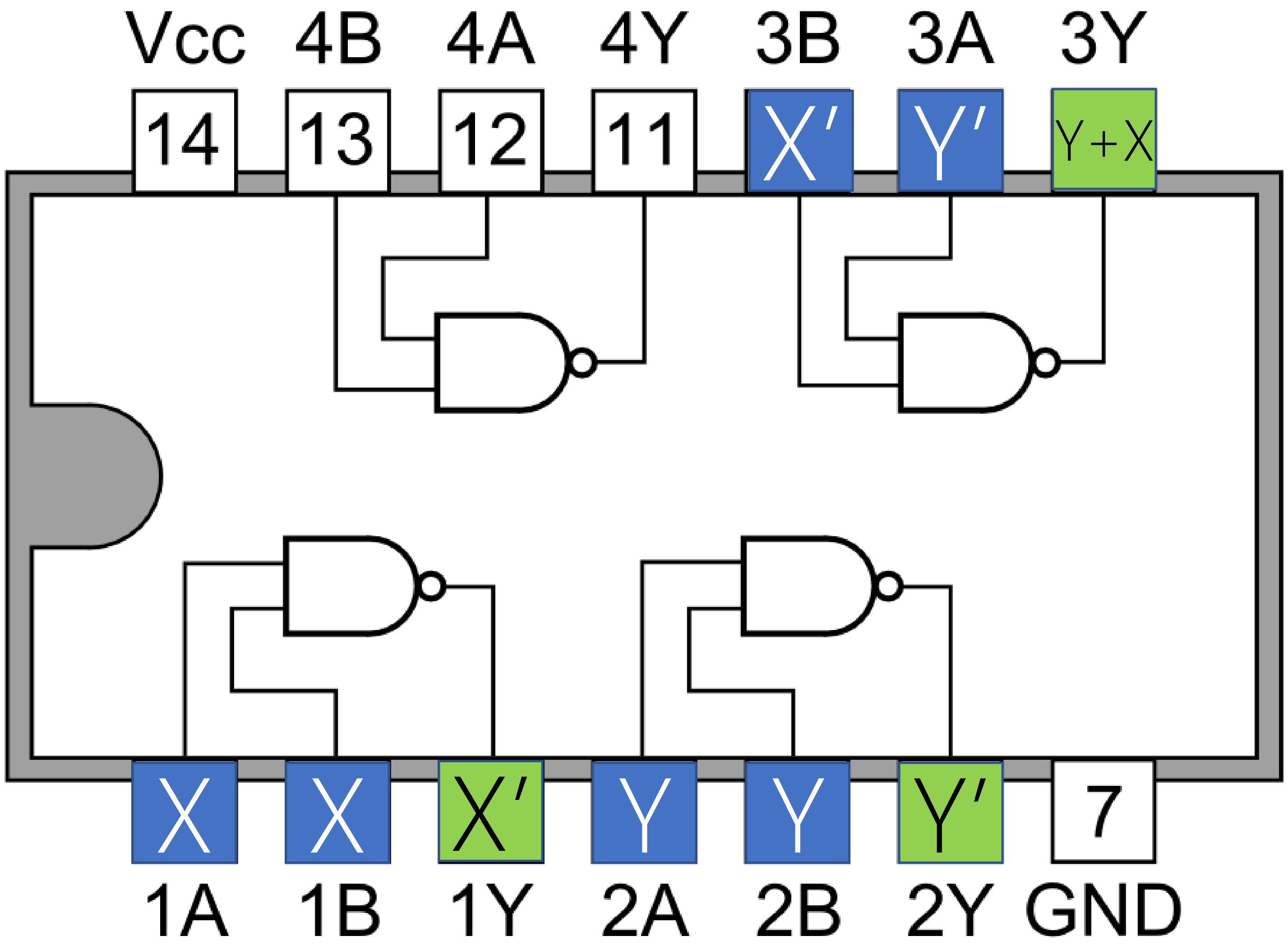
7400 Quad 2-input NAND Gates

https://commons.wikimedia.org/wiki/7400_series_overview





7400 Quad 2-input NAND Gates



SET	UNIVERSAL SET
{NOT, AND, OR, NAND, NOR}	Yes! (a Full Set)
{NOT, AND, OR}	Yes, (a Full Set) NAND = AND-NOT NOR = OR-NOT
{NOT, AND}	If we could design OR (a Complete Set)
{NOT, OR}	If we could design AND (a Complete Set)
{NOT}	If we could design AND, OR
{AND}	If we could design NOT, OR
{OR}	If we could design NOT, AND
{NAND}	If we could design NOT, AND, OR
{NOR}	If we could design NOT, AND, OR

UNIVERSAL GATE
{NOR}

NOT ► $(X+X)' = (XX) = X'$

OR ► NOT (NOR)

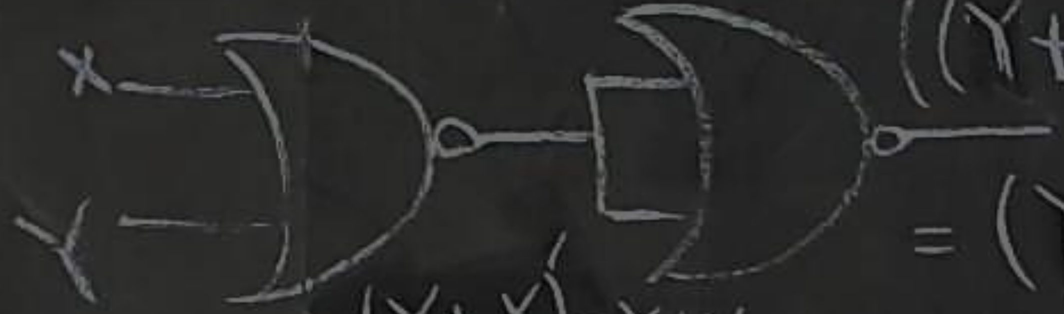
AND ► DE MORGAN'S LAW

$(Y'+X')' = YX = (Y'{\downarrow}X')$

NOR

NOT

OR



$(Y+X)' = Y \downarrow X$

$((Y+X)')'$

$= (Y+X)$

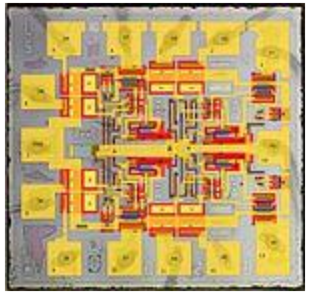
AND

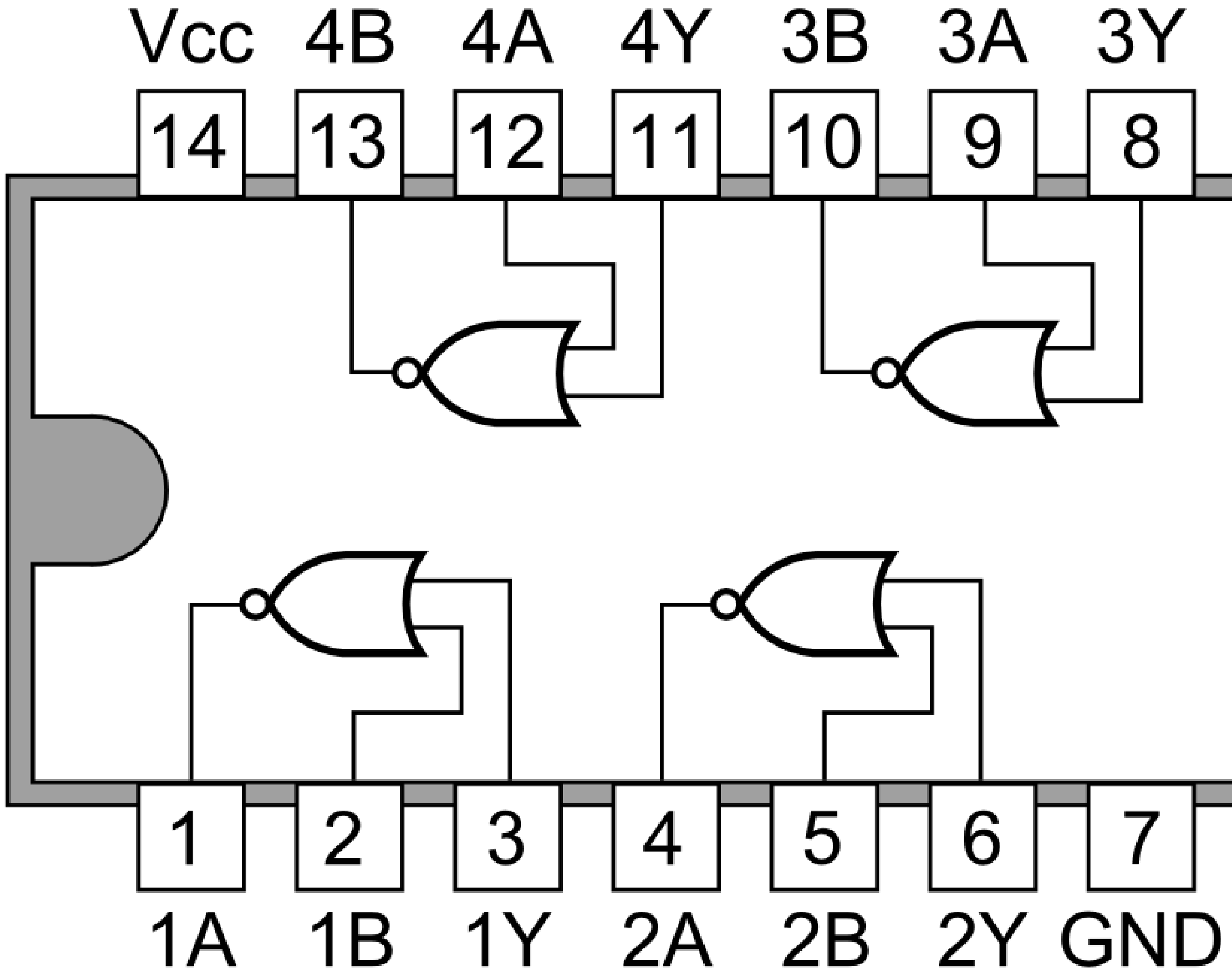
$(Y'+X')' = Y' \downarrow X'$

$= YX$

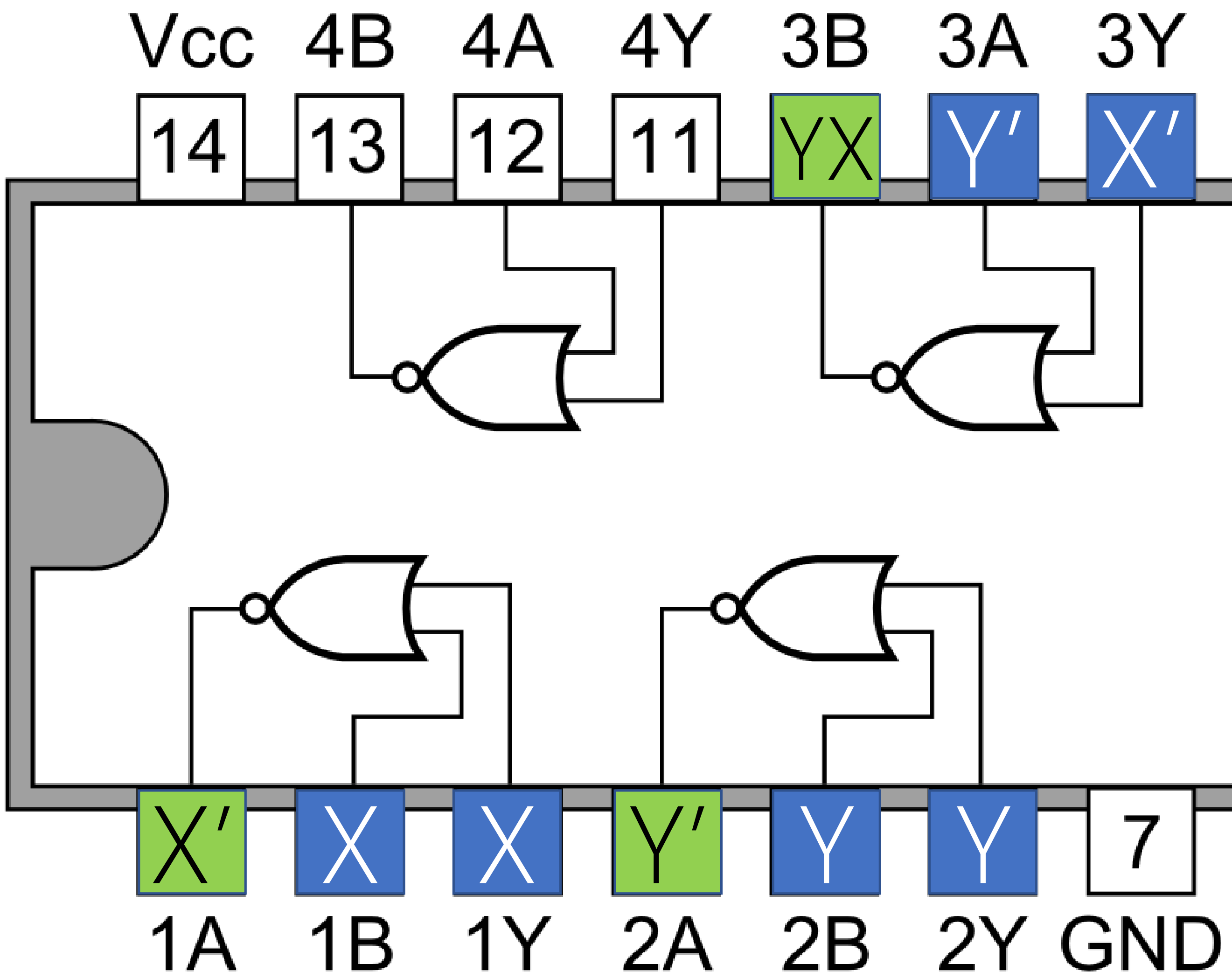
7402 Quad 2-input NOR Gates

https://commons.wikimedia.org/wiki/7400_series_overview





7402 Quad 2-input NOR Gates



DESIGN

a design algorithm for any digital units (logic circuits), given truth table

minterm

aka. Standard Product

X′ vs. X

1 binary variable appear either:

- in its normal form X, or
- in its complement form X′

m_0	X'
m_1	X

YX vs. YX' vs. $Y'X$ vs. $Y'X'$

2 binary variables appear either in one of these forms:

m_0	Y'X'
m_1	Y'X
m_2	YX'
m_3	YX

ZYX vs. ZYX’ vs. …

3 binary variables appear either in one of these forms: how many?

ZYX vs. ZYX' vs. …

3 binary variables appear either in one of these forms: how many?
Each variable can take 2 forms (normal and complement)
We have 3 variables, $2 \times 2 \times 2 = 2^3 = 8$

m_0	$Z'Y'X'$
m_1	$Z'Y'X$
m_2	$Z'YX'$
m_3	$Z'YX$
m_4	$ZY'X'$
m_5	$ZY'X$
m_6	ZYX'
m_7	ZYX

$$A_n \cdots A_2 A_1 A_0 \text{ vs. } A_n \cdots A_2 A_1 A'_0 \ldots$$

n binary variables appear either in one of these forms: how many?

Each variable can take 2 forms (normal and complement)

We have n variables, $2 \times 2 \times 2 \times \cdots \times 2 = 2^n$

m_0	$A'_n \cdots A'_2 A'_1 A'_0$
m_1	$A'_n \cdots A'_2 A'_1 A_0$
m_2	$A'_n \cdots A'_2 A_1 A'_0$
$\vdots$	$\vdots$
$\vdots$	$\vdots$
m_{2^n-3}	$A_n \cdots A_2 A'_1 A_0$
m_{2^n-2}	$A_n \cdots A_2 A_1 A'_0$
m_{2^n-1}	$A_n \cdots A_2 A_1 A_0$

TRUTH TABLE

en.wikipedia.org/wiki/Truth_table

X	F = F(X) = ?
0	?
1	?

X	F = F(X) = 1
0	1
1	1

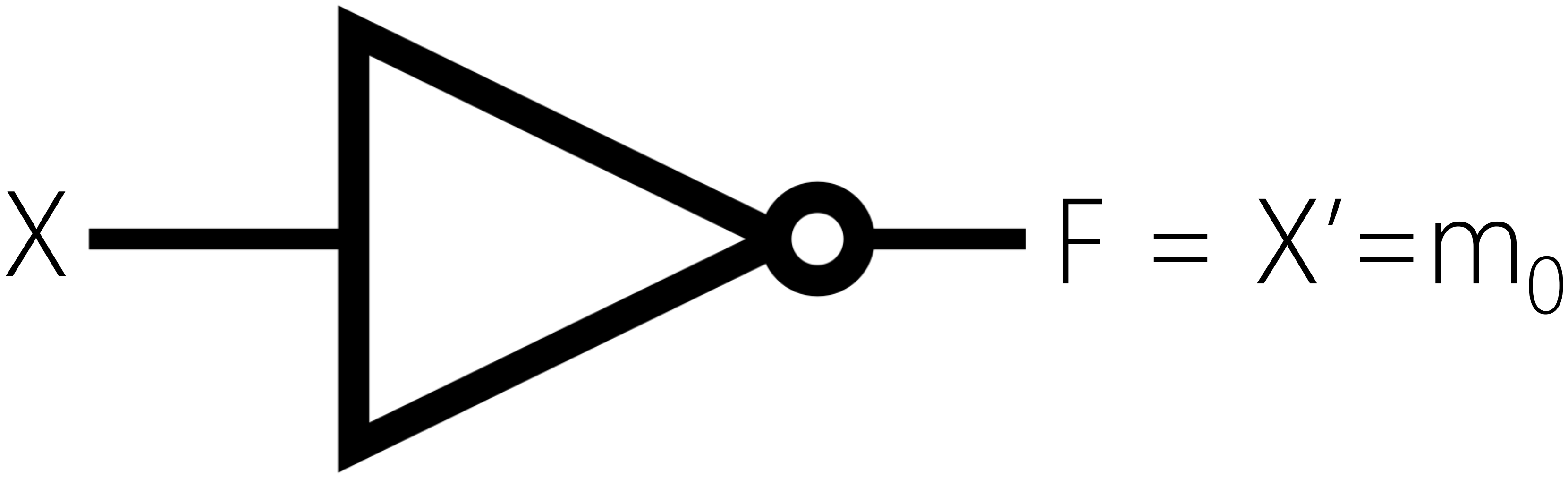
X	F = F(X) = 0
0	0
1	0

X	F = F(X) = X′
0	1
1	0

m_0	X'
m_1	X

X	$F = F(X) = X' = m_0$
0	1
1	0

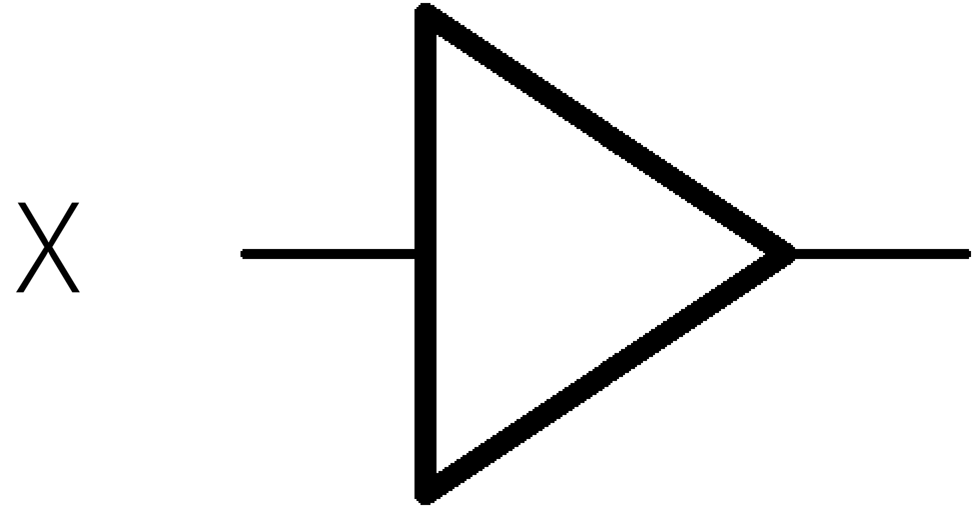
X	$F = F(X) = X' = m_0$
0	1
1	0


X
F = X’=m0

X	F = F(X) = X
0	0
1	1

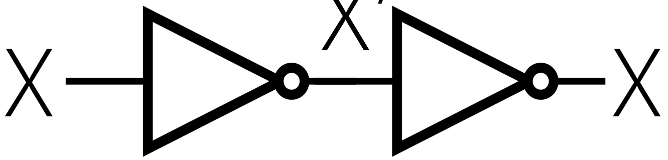
X	$F = F(X) = X = m_1$
0	0
1	1

X	F = F(X) = X = m_1
0	0
1	1



$F = X = m_1$

Digital Buffer



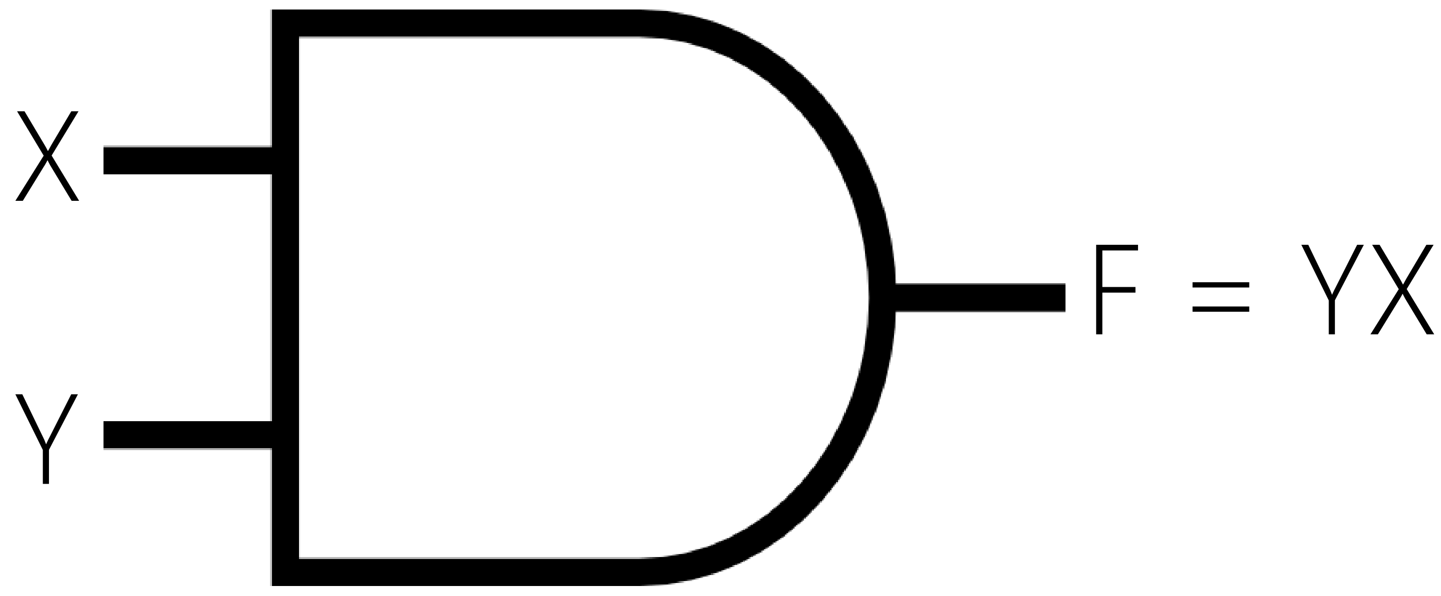
TRUTH TABLE ↔ minterm

Y	X	F = F(Y,X) = ?
0	0	?
0	1	?
1	0	?
1	1	?

Y	X	$F = F(Y,X) = 0$
0	0	0
0	1	0
1	0	0
1	1	0

Y	X	F = F(Y,X) = 1
0	0	1
0	1	1
1	0	1
1	1	1

Y	X	F = F(Y,X) = YX
0	0	0
0	1	0
1	0	0
1	1	1


X
Y
F = YX

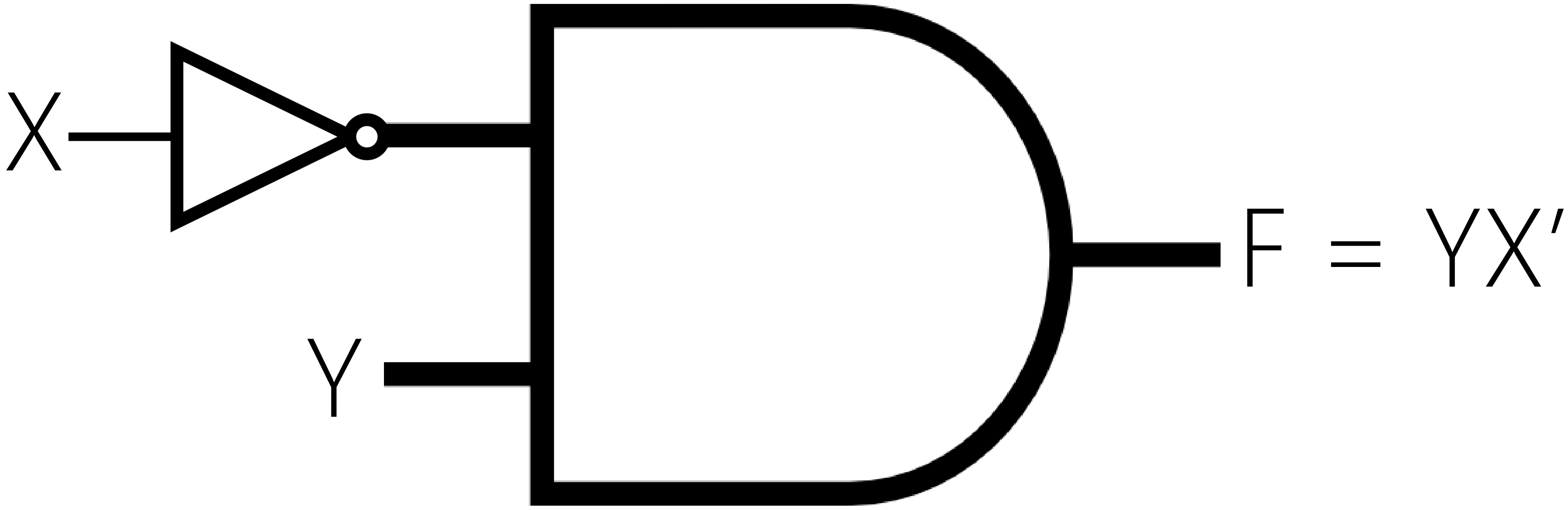
m_0	$Y'X'$
m_1	$Y'X$
m_2	YX'
m_3	YX

Y	X	$F = F(Y,X) = YX = m_3$
0	0	0
0	1	0
1	0	0
1	1	1

Y	X	$F = F(Y,X) = YX = m_3$
0	0	0
0	1	0
1	0	0
1	1	1

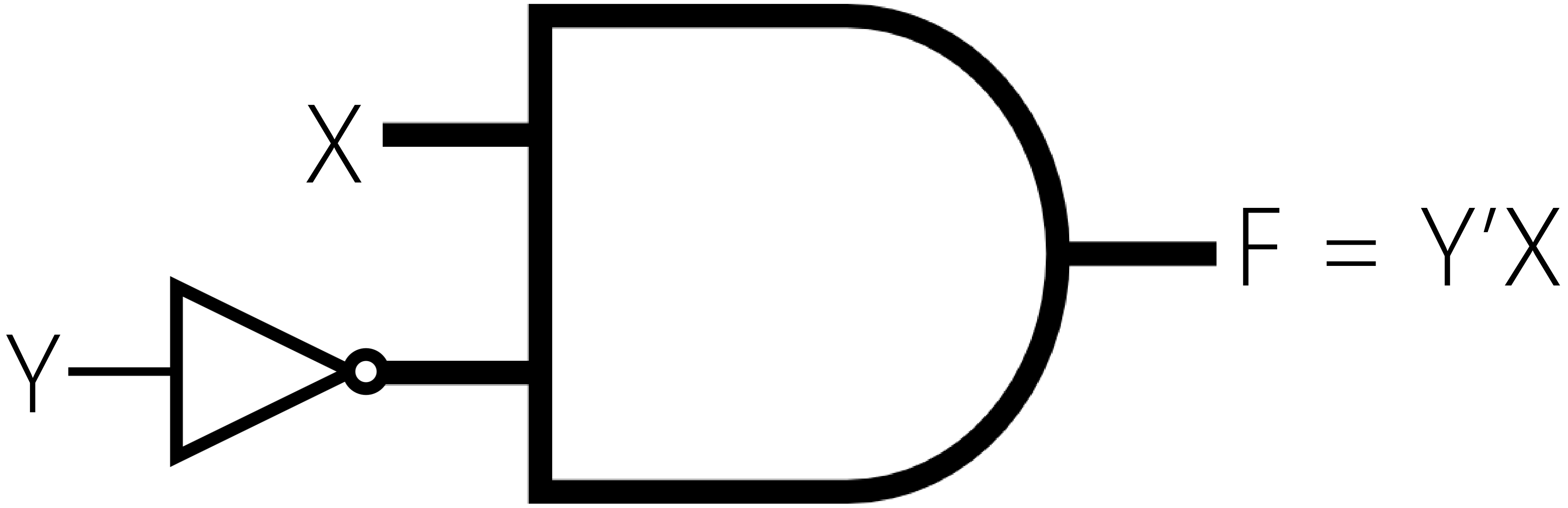
Y	X	$F = F(Y,X) = YX' = m_2$
0	0	0
0	1	0
1	0	1
1	1	0

Y	X	$F = F(Y,X) = YX' = m_2$
0	0	0
0	1	0
1	0	1
1	1	0


X
Y
F = YX'

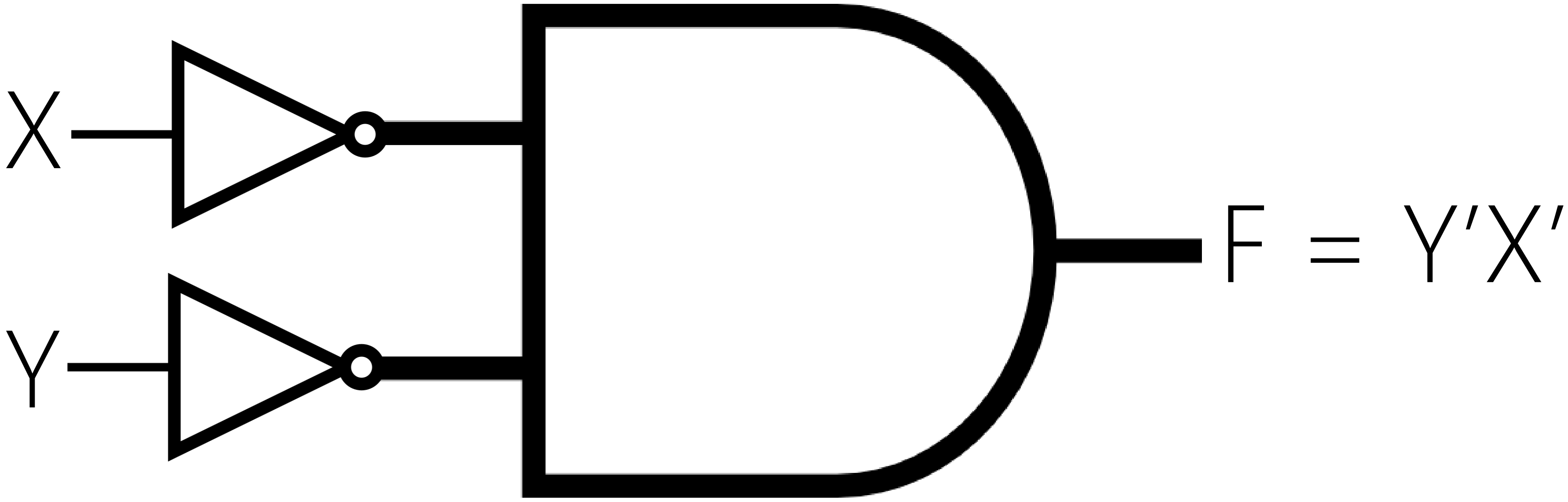
Y	X	$F = F(Y,X) = Y'X = m_1$
0	0	0
0	1	1
1	0	0
1	1	0

Y	X	$F = F(Y,X) = Y'X = m_1$
0	0	0
0	1	1
1	0	0
1	1	0


X
Y
F = Y'X

Y	X	$F = F(Y,X) = Y'X' = m_0$
0	0	1
0	1	0
1	0	0
1	1	0

Y	X	$F = F(Y,X) = Y'X' = m_0$
0	0	1
0	1	0
1	0	0
1	1	0


X
Y
F = Y′X′

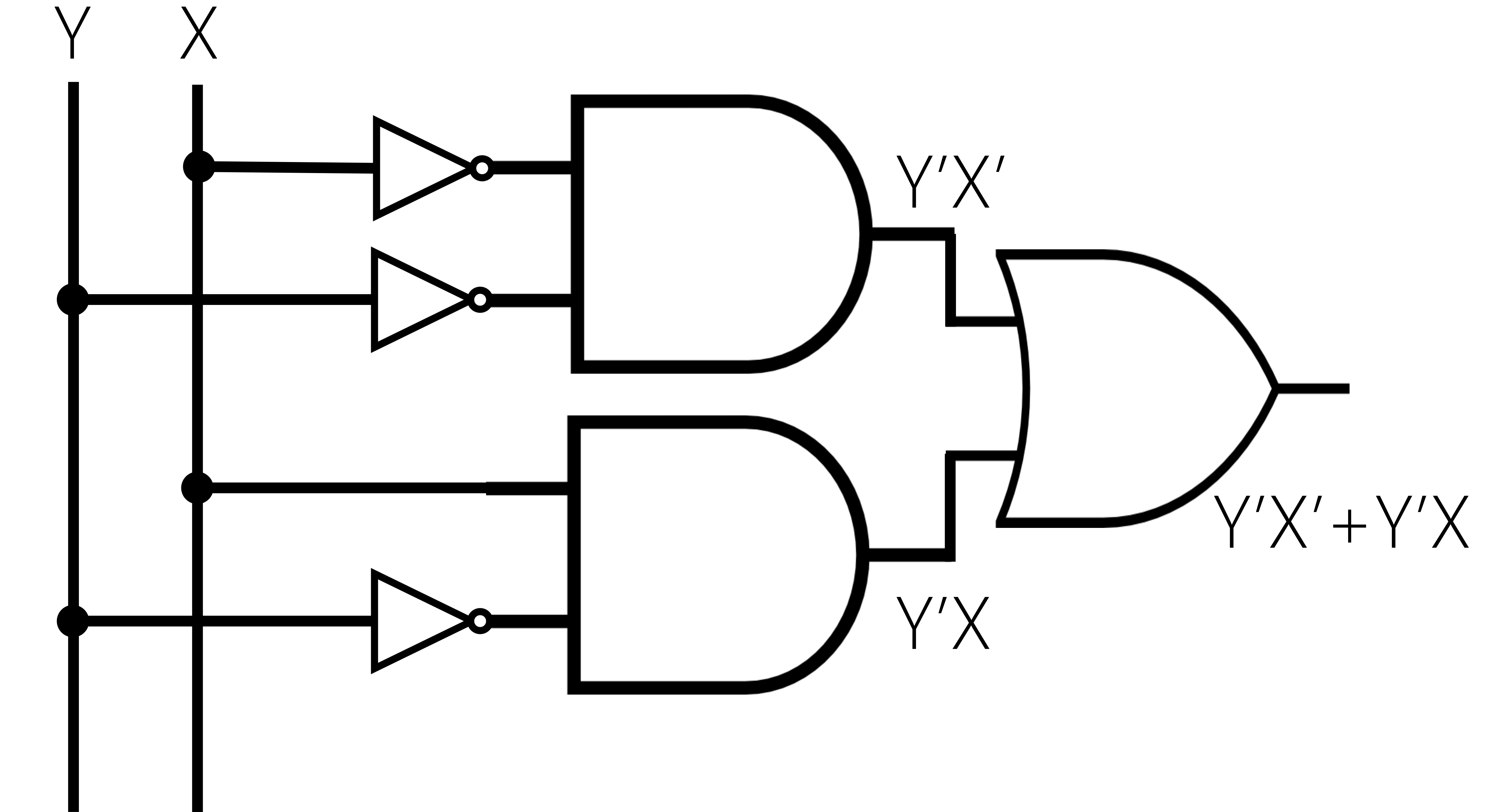
Y	X	F = F(Y,X) = ?
0	0	1
0	1	1
1	0	0
1	1	0

Y	X	F = F(Y,X) = Y’X’
0	0	1
0	1	1
1	0	0
1	1	0

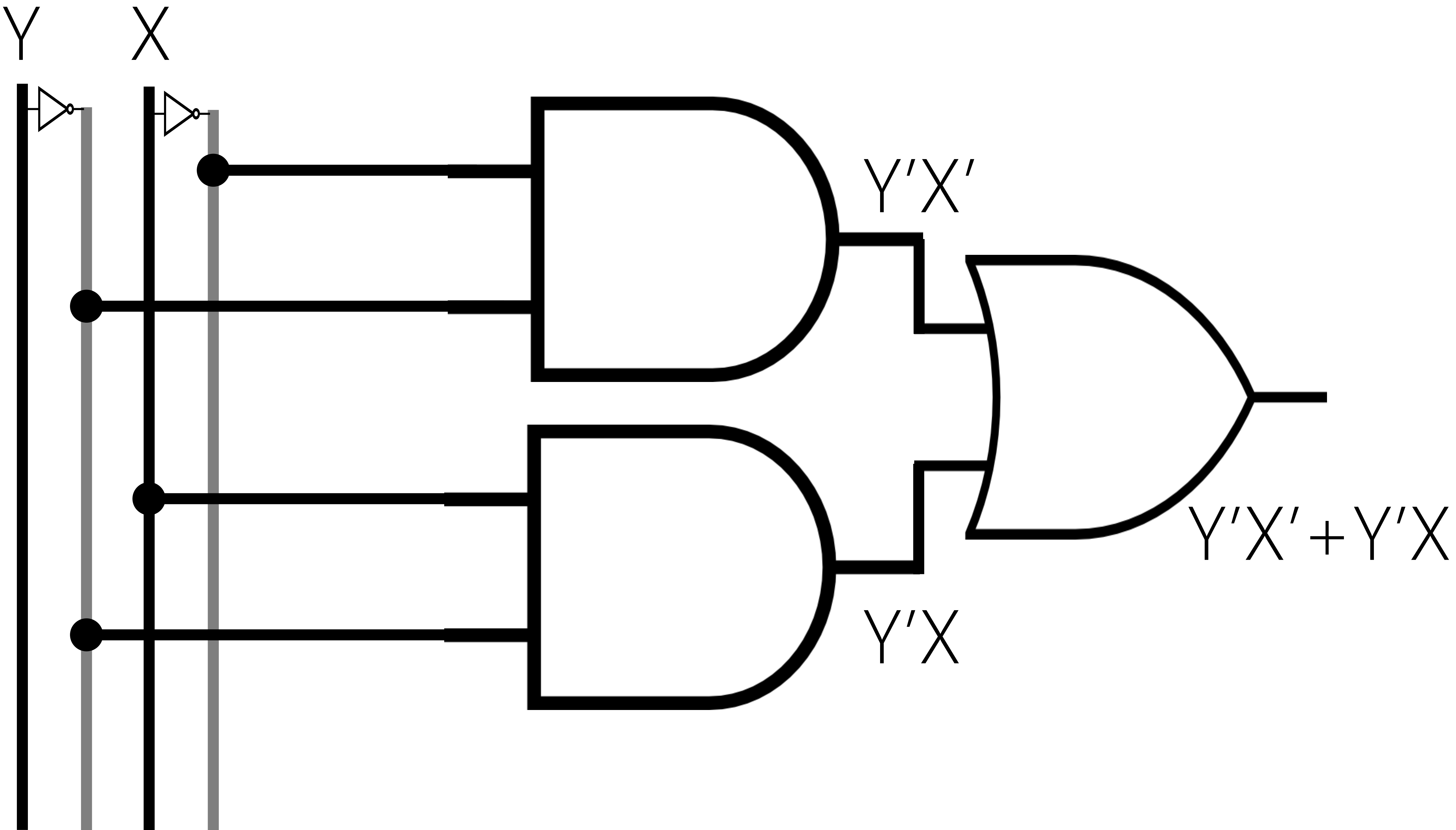
Y	X	F = F(Y,X) = Y'X'+Y'X
0	0	1
0	1	1
1	0	0
1	1	0

Y	X	$F = F(Y,X) = m_0+m_1$
0	0	1
0	1	1
1	0	0
1	1	0

Y	X	$F = F(Y,X) = \sum m(0,1)$
0	0	1
0	1	1
1	0	0
1	1	0



Y
X
Y′X′
Y′X
Y′X′+Y′X

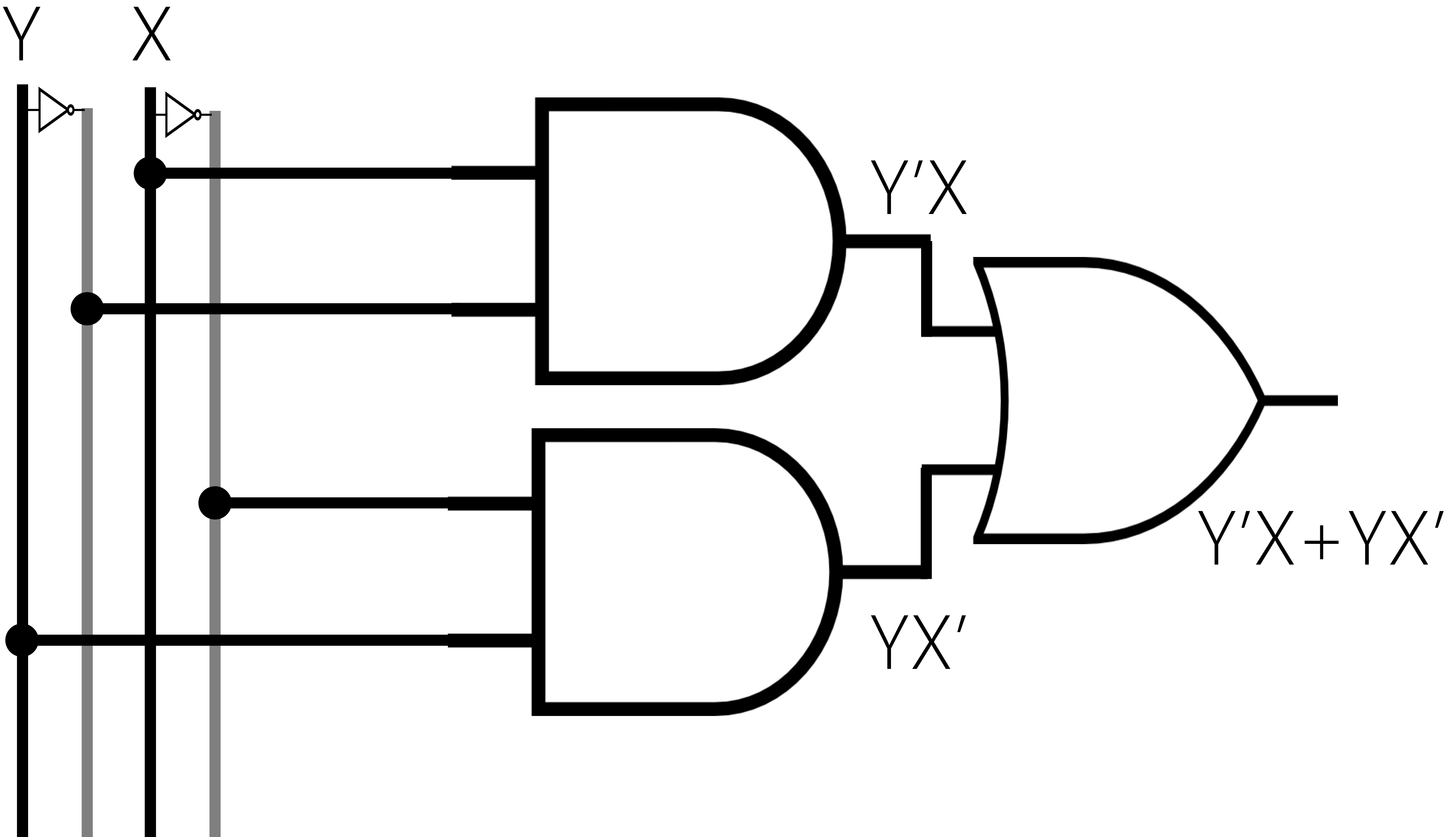

Y
X
Y′X′
Y′X
Y′X′+Y′X

Y	X	F = F(Y,X) = ?
0	0	0
0	1	1
1	0	1
1	1	0

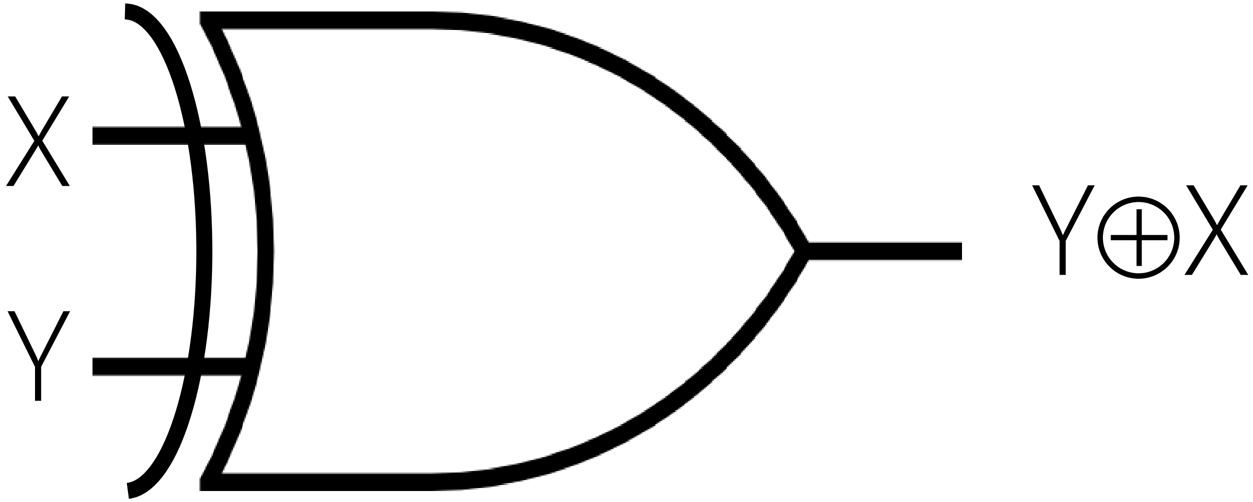
Y	X	F = F(Y,X) = Y'X
0	0	0
0	1	1
1	0	1
1	1	0

Y	X	$F = F(Y,X) = m_1+m_2$
0	0	0
0	1	1
1	0	1
1	1	0

Y	X	$F = F(Y,X) = \sum m(1,2)$
0	0	0
0	1	1
1	0	1
1	1	0


Y
X
Y’X
YX’
Y’X+YX’

Exclusive-OR (XOR)



Y	X	F = F(Y,X) = Y'X+YX' = m_1+m_2
0	0	0
0	1	1
1	0	1
1	1	0

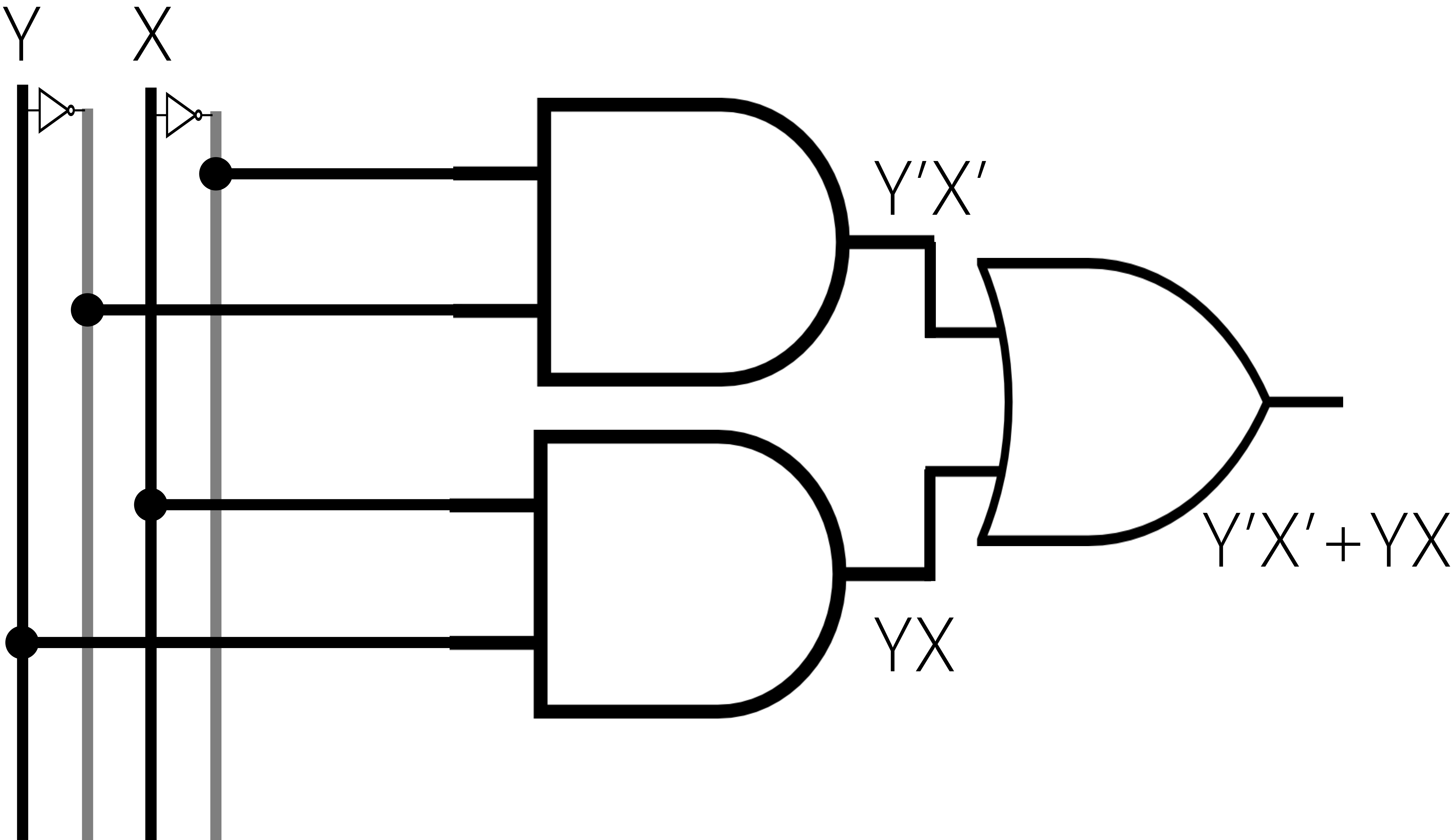
Y	X	F = F(Y,X) = ?
0	0	1
0	1	0
1	0	0
1	1	1

Y	X	F = F(Y,X) = Y’X’
0	0	1
0	1	0
1	0	0
1	1	1

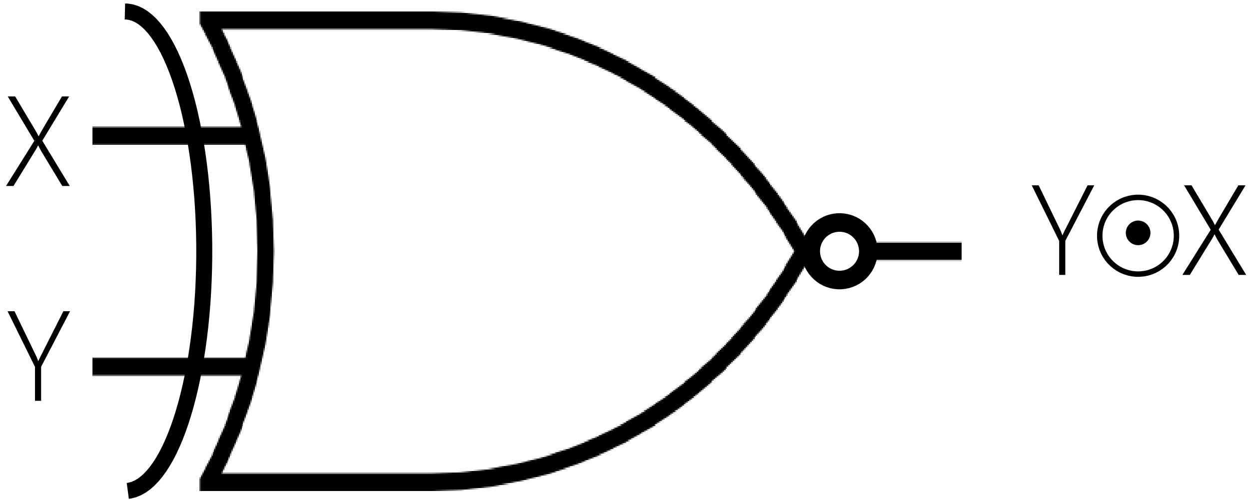
Y	X	$F = F(Y,X) = Y'X'+YX$
0	0	1
0	1	0
1	0	0
1	1	1

Y	X	$F = F(Y,X) = m_0 + m_3$
0	0	1
0	1	0
1	0	0
1	1	1

Y	X	$F = F(Y,X) = \sum m(0,3)$
0	0	1
0	1	0
1	0	0
1	1	1


Y
X
Y′X′
YX
Y′X′+YX

NOT Exclusive-OR (XNOR)



Y	X	F = F(Y,X) = Y'X'+YX = m_0+m_3
0	0	1
0	1	0
1	0	0
1	1	1

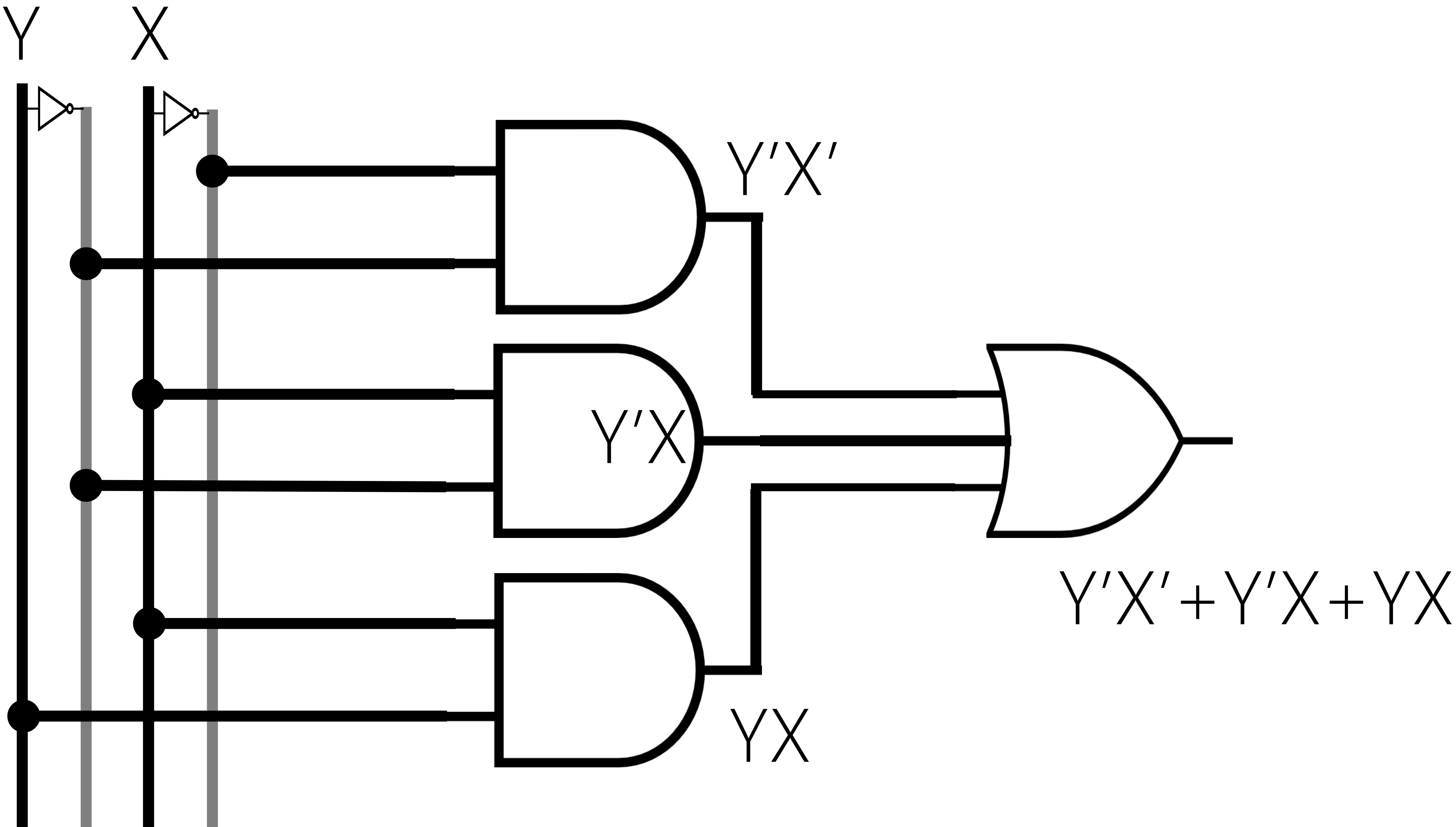
Y	X	F = F(Y,X) = ?
0	0	1
0	1	1
1	0	0
1	1	1

Y	X	F = F(Y,X) = Y'X'
0	0	1
0	1	1
1	0	0
1	1	1

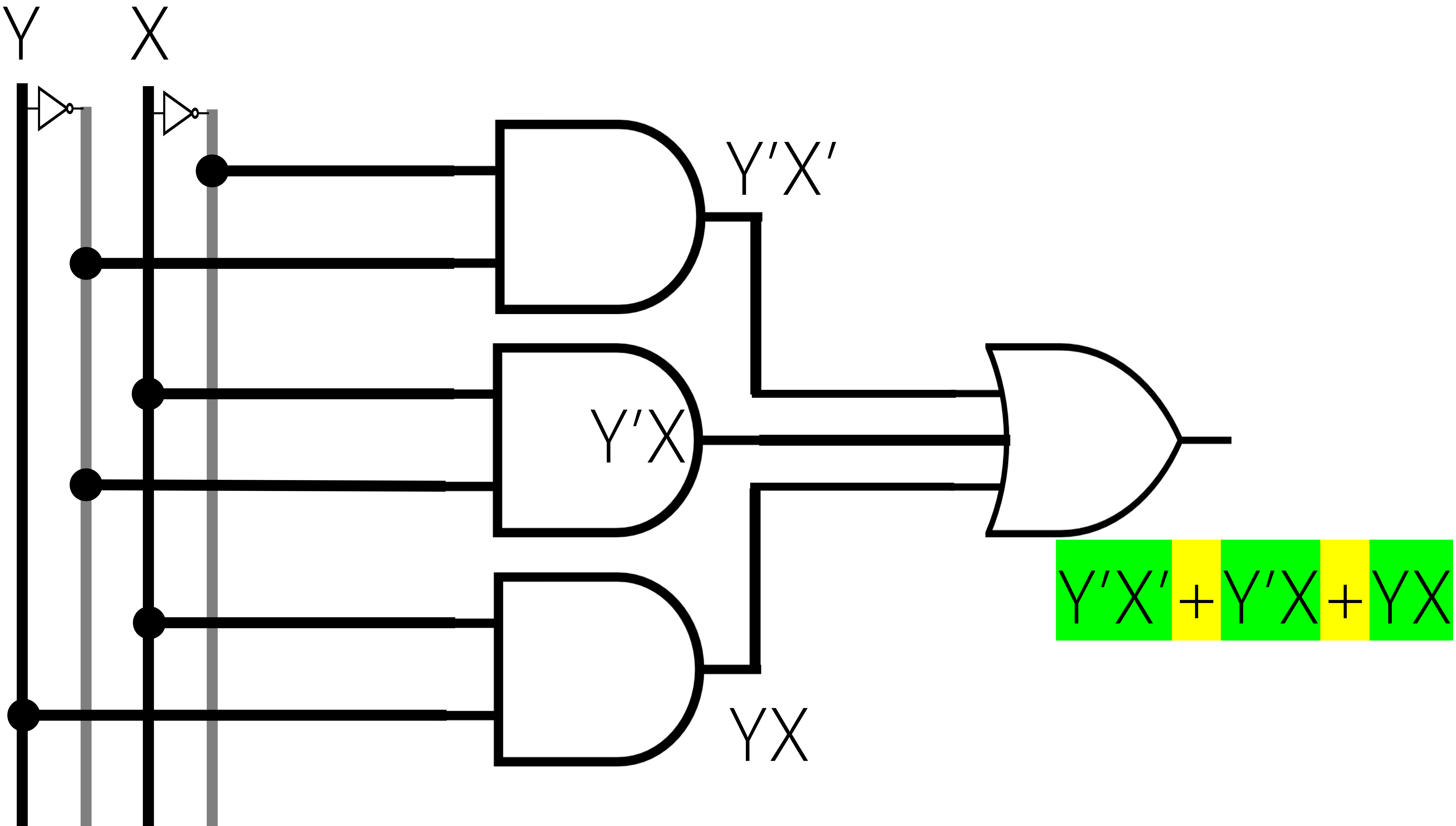
Y	X	$F = F(Y,X) = Y'X' + Y'X$
0	0	1
0	1	1
1	0	0
1	1	1

Y	X	F = F(Y,X) = Y'X'+Y'X+YX
0	0	1
0	1	1
1	0	0
1	1	1

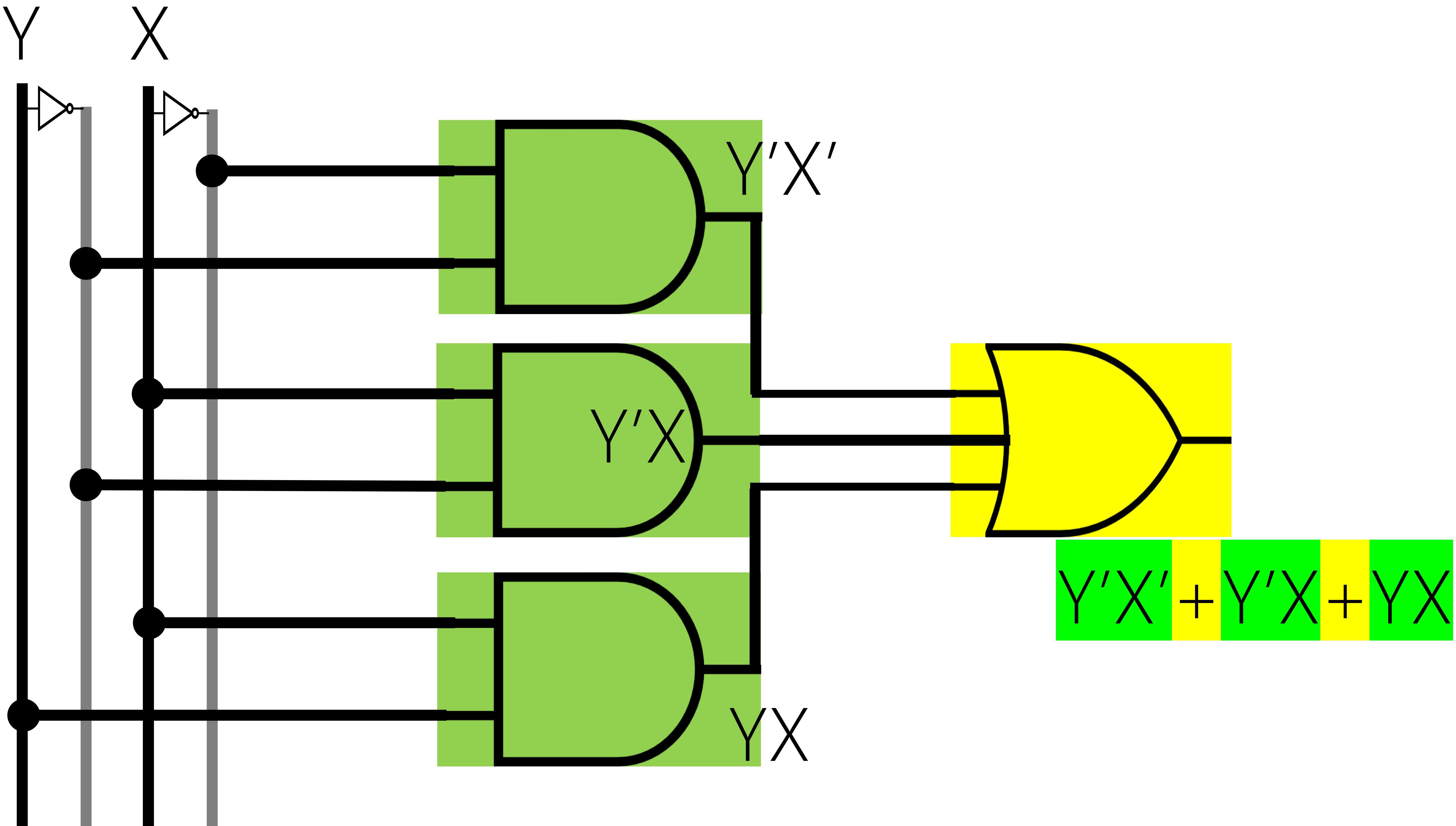
Y	X	$F = F(Y,X) = m_0 + m_1 + m_3 = \sum m(0,1,3)$
0	0	1
0	1	1
1	0	0
1	1	1


Y
X
Y′X′
Y′X
YX
Y′X′+Y′X+YX

SUM OF PRODUCTS (SOP)


Y
X
Y'X'
Y'X
YX
Y'X'+Y'X+YX

2 LEVELS
AND → OR



Y
X
Y'X'
Y'X
YX
Y'X'+Y'X+YX

Given 3 inputs, design a circuit to determine if there is even number of 1

Z	Y	X	F(Z,Y,X)=?
0	0	0	?
0	0	1	?
0	1	0	?
0	1	1	?
1	0	0	?
1	0	1	?
1	1	0	?
1	1	1	?

Z	Y	X	F(Z,Y,X)=?
0	0	0	1
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	0

Z	Y	X	F(Z,Y,X)=Z′Y′X′
0	0	0	1
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	0

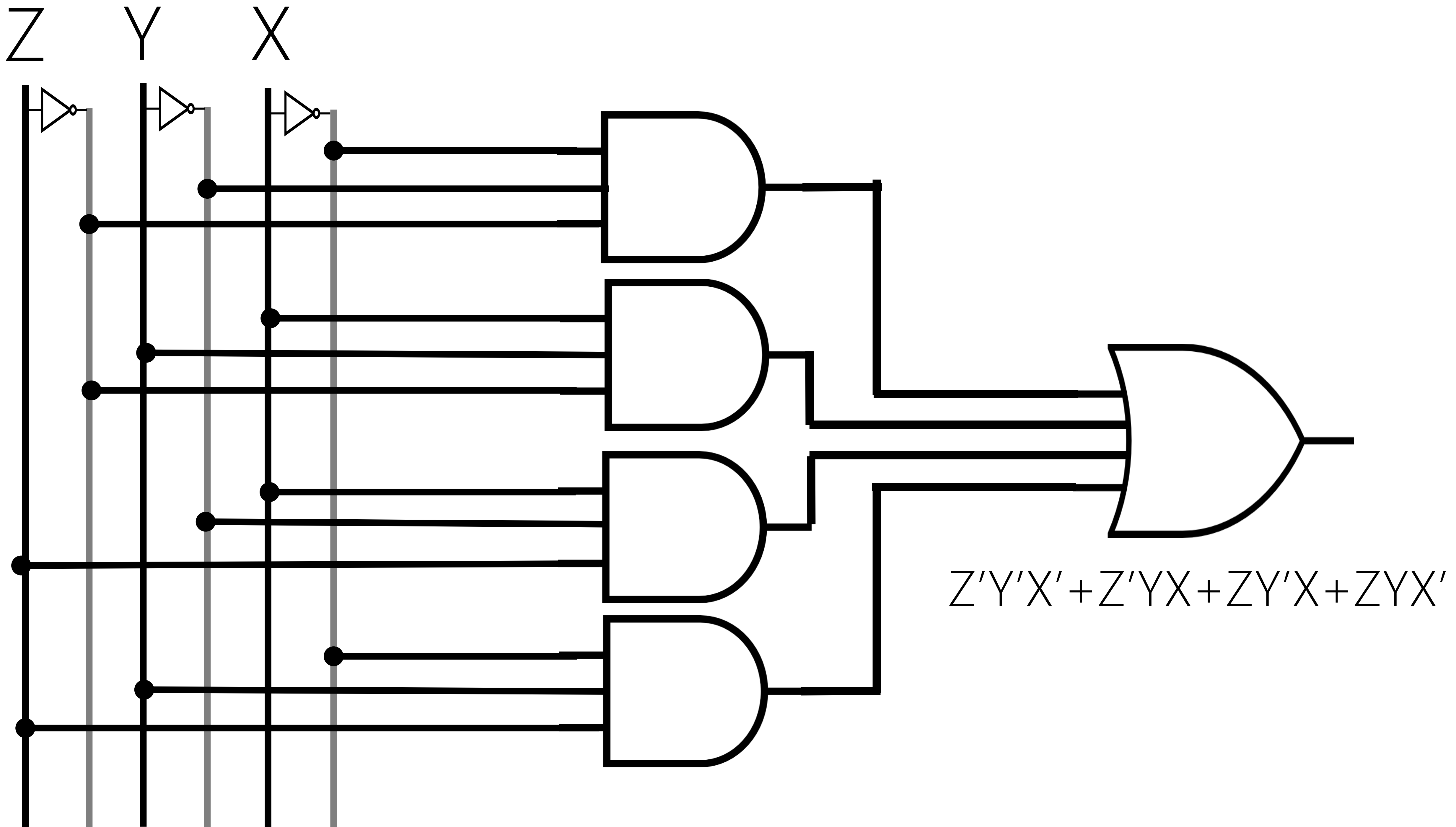
Z	Y	X	$F(Z,Y,X)=Z'Y'X'+Z'YX$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Z	Y	X	$F(Z,Y,X)=Z'Y'X'+Z'YX+ZY'X$
0	0	0	1
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	0

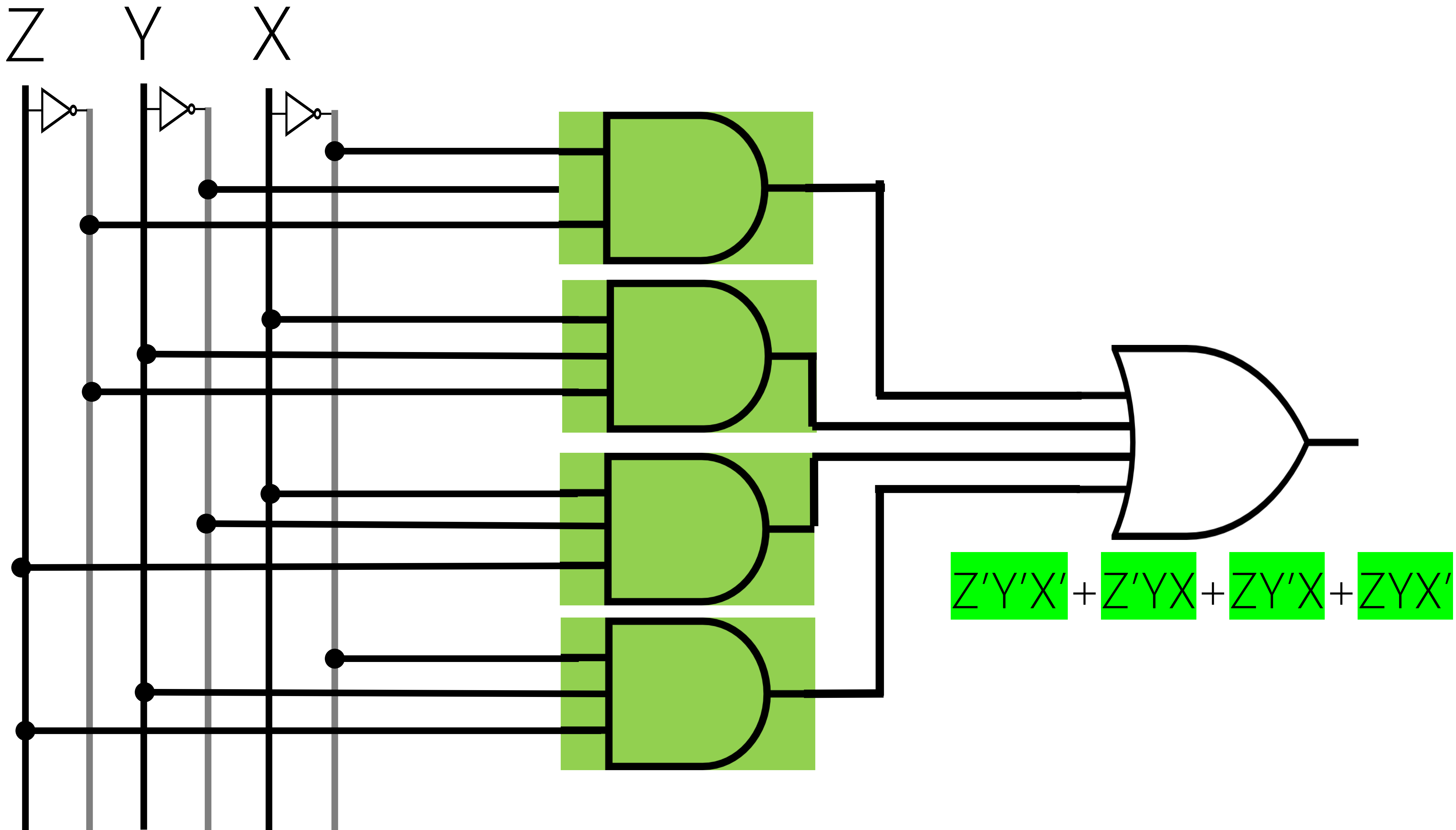
Z	Y	X	$F(Z,Y,X)=Z'Y'X'+Z'YX+ZY'X+ZYX'$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

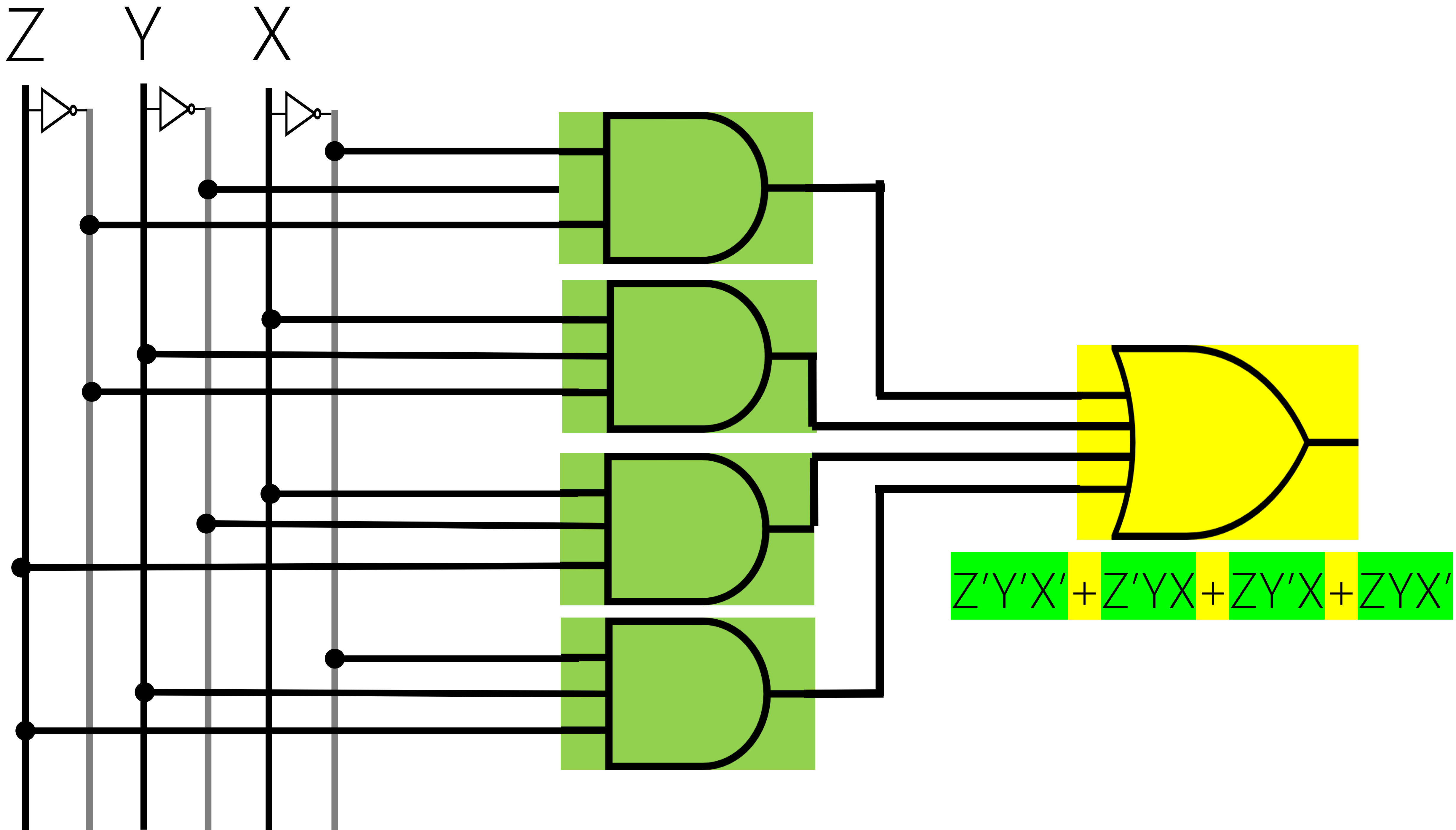
Z	Y	X	$F(Z,Y,X)=m_0+m_3+m_5+m_6$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Z	Y	X	$F(Z,Y,X)=m_0+m_3+m_5+m_6=\sum m(0,3,5,6)$
0	0	0	1
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	0


Z
Y
X
Z'Y'X'+Z'YX+ZY'X+ZYX'

SUM OF PRODUCTS (SOP)
2 LEVELS AND-OR


Z
Y
X
Z'Y'X' + Z'YX + ZY'X + ZYX'


Z
Y
X
Z'Y'X' + Z'YX + ZY'X + ZYX'